

Die Datenbank MySQL als Hilfsmittel für
die Darstellung geschichtlicher Themen

von

Heinz Wember

Die Datenbank MySQL als Hilfsmittel für die Darstellung geschichtlicher Themen

Heinz Wember

2. Auflage (php 7.2)

Erscheinungsjahr: 2020

Erscheinungsort: Augsburg

Im Selbstverlag

Änderungsstand: 20-Mai-2020

Datei: F:\Wember\Heinz\BuchMySQL2Auflage.indd

Druck: Druckerei der Universität Augsburg

ISBN 978-3-00-065480-0



Die Datenbank MySQL als Hilfsmittel für die Darstellung geschichtlicher Themen

Heinz Wember

Inhaltsverzeichnis

Inhaltsverzeichnis	4
Vorwort	5
Technische Voraussetzungen	7
Systematik Navigation	13
Beispiele für Select-Befehle	14
Relative oder fixe Spaltenbreite	17
Beispiel: Textgestaltung	19
Interaktive Abfrage	21
Erstellung von Seiten	24
Darstellung des Datums	26
Datenbank Design	33
Kommentare zum Password	42
Tipps zur Programmierung und zum Testen der Programme	45
Teamviewer und SFTP	47
Quellen und Literatur	48
Wie wurde dieses Buch geschrieben?	49
Nachwort	50
Anhang	51-137
Examples documented in english	138-149
Über den Autor	150

Vorwort

Diese Dokumentation richtet sich natürlich nicht an Informatiker, die dies beherrschen (sollten), sondern an Geisteswissenschaftler, die ein Thema mit Hilfe der Datenbank MySQL im Internet oder auch lokal auf einem PC (evtl. für einen Vortrag mit PC mit Übertragung auf eine Leinwand) darstellen wollen.

Ausgelöst wurde dieses Thema dadurch, dass ich nach 50jähriger EDV-Erfahrung nun als Hobby die Geschichte entdeckt habe. Natürlich sind diese Methoden, die ich hier darstelle, nicht nur auf das Fach Geschichte beschränkt, man kann genauso gut Themen der Medizin, Betriebswirtschaft etc. damit darstellen.

Nachdem ein relativ umfangreiches Beispiel über ein heimatgeschichtliches Thema bereits im Internet verfügbar ist, sollen einige dieser Methoden auch für andere Interessenten dargestellt werden.

Nachdem ich erkannt hatte, dass man mein heimatgeschichtliches Thema (Die Geschichte von Augsburg-Kriegshaber) nicht so ohne weiteres in einem Buch kostengünstig darstellen kann, habe ich zu diesen Werkzeugen (MySQL, php, html) gegriffen. Zum einem ist für viele wissenschaftliche Themen der Kreis der Leser so eines umfangreichen Buches relativ klein, d.h. ein Buch kann nicht kostendeckend herausgebracht werden, selbst wenn man dies im Eigenverlag macht. In meinem Beispiel Geschichte von Kriegshaber wollte ich den historischen Kataster unseres Ortes relativ komplett darstellen. Momentan ist der Umfang der Daten dieser Datenbank über 4 Millionen Byte, es würde also ein Buch von mehreren tausend Seiten ergeben. Andererseits sind die Möglichkeiten des Suchens nach Begriffen, Orten und Personennamen in einem Buch einigermaßen beschränkt. Will man z.B. alle Häuser, die im Kataster erfasst sind, darstellen, genügt es in der Regel nicht, dies in nur einer Sortierfolge zu tun. Hier sind Sortierfolgen nach verschiedenen Begriffen wie Flurnummer, Hausnummer, Besitzernamen sehr hilfreich. Dies bläht die Seitenzahl entsprechend auf. Stehen einmal alle Häuser in einer Datenbanktabelle, ist es ein Leichtes, diese Daten interaktiv zu sortieren. Auch ist in der Regel der Leser nur an bestimmten Auswahlbegriffen interessiert. Dies lässt sich mit einem Begriff, den man interaktiv eingeben kann, leicht lösen. Natürlich setzt dies ein bestimmtes EDV-Wissen voraus, das aber bei weitem nicht so detailliert sein muss, wie dies in der Regel ein Informatiker hat. Ein konkretes Beispiel ist oft ausreichend, um die Prinzipien sinnvoll auch für andere Themen nutzbar zu machen. Ein weiterer Vorteil einer Datenbank ist die Möglichkeit, die Daten redundanzfrei speichern zu können. Nachdem z.B. der Straßename von Häusern sehr oft gleich ist, in Kriegshaber z.B. gehen die Hausnummern der Häuser in der Ulmer Straße bis über 300 hinaus, ist es nutzbringend, nicht jeweils den kompletten Straßennamen in der Tabelle Häuser zu speichern, sondern nur die Nummer der Straße. Der Textname der Straße kann bei Aufruf eines Reports automatisch hinzugefügt werden. Damit ist natürlich auch der Pflegeaufwand, wenn sich solche Daten ändern, entsprechend gering. Der Anfänger wird wohl am Anfang nur die Tools benutzen, die zur Erfassung der Daten in einer speziellen Tabelle gebraucht werden, später wird er auch die Vorteile der Pflegebefehle wie Ändern und Kopieren von Daten sinnvoll anwenden können, besonders wenn dies durch vorhandene Musterbeispiele erleichtert wird. Das Lesen dicker Manuale ist nicht jedermanns Sache. Nachdem es mir nicht gelungen ist, einen Nachfolger für meine Homepage zu bekommen, habe ich inzwischen die wichtigsten Ergebnisse der Homepage zu Kriegshaber auch als Buch herausgegeben (leider hat das Buch 3600 Seiten). Nachdem der Druck so teuer ist, wird das Buch nur in den wichtigsten Bibliotheken zu finden sein.

Darstellung auf einem lokalen PC bzw. kleinem Netzwerk und im Internet

Was hier gesagt wird, bezieht sich nicht nur auf die Darstellung auf einer Homepage im Internet, die ohnehin entweder öffentlich oder mit Passwort geschützt sein kann, sondern auch auf die Darstellung in einem kleinen Netzwerk (Abteilung, Institut) oder auf einem persönlichen Laptop. Ein weiterer Vorteil dieser Arbeitsweise ist der, dass man durch eine minimale Änderung eines einzigen Aufrufskriptes die Daten sowohl im Internet als auch lokal haben kann. Dies habe ich intensiv dazu benutzt, dass ich die Datenerfassung im Archiv direkt in die MySQL-Datenbank auf meinem Laptop benutzte, zu Hause nach entsprechender Korrekturlesung diese Daten exportierte und auf meine Homepage importieren konnte. Die Programme auf dem Laptop und im Internet sind identisch, anders ist nur ein Skript, das man bei der Anmeldung für die spezielle Anwendung braucht.

Dies ist nun die 2. Auflage dieses Buches. Die erste Auflage hatte noch die Release 5.3 für die PHP-Programme. Inzwischen wird diese Version bei den Providern (bei mir lund1.de bzw. ionos.de) nicht mehr unterstützt. 2019 habe ich alle meine Programme auf die gegenwärtige PHP-Version 7.2 umgestellt. Nachdem hierbei der Zugang zu der MySQL-Datenbank geändert wurde, musste ich alle Programme ändern. Aber es waren nur wenige Befehle, die in der Release 7.2 neu waren bzw. geändert wurden. Im Anhang gibt es eine Gegenüberstellung dieser Befehle von PHP-Version 5.3 zu PHP-Version 7.2. Die aktuelle PHP-Version im Internet ist 7.2.287.2.28. Die Release von MySQL 5.5.60-0+deb7u1 blieb unverändert.

Danken möchte ich meiner Schwester Gisela Wember, die das Konzept Korrektur gelesen hat.

Technische Voraussetzungen

1. Die Datenbank MySQL und der Apache-Server

Diese Datenbank ist sowohl für die Homepage als auch für den Gebrauch unter Windows auf einem Laptop kostenlos. Dazu braucht man auch einen Apache-Server, der ebenfalls kostenlos heruntergeladen werden kann. Hat man Windows, so ist eine Option die Verwendung des kostenlosen Xampp-Programms. Im Internet benutze ich als Provider lundl.de, die einen Weospace für eine private Homepage mit einer MySQL-Datenbank für ca. 7 € Monatsmiete anbietet. Größere Anwendungen sind natürlich teurer, ich persönlich habe eine eigene CPU mit einem Weospace von 140 GB. Dies kostet pro Monat dann etwa 40 €. Wenn man bedenkt, wie teuer das Herausgeben eines Buches von etwa 200 Seiten kommen kann, muss man hier je nach Papierqualität und die Art, wie das Buch gebunden wird, schon mit mehreren tausend Euro für eine kleine Auflage von etwa 100 Stück rechnen.

2. Das Erstellen der Programme

Damit die Daten der Datenbank auf dem Bildschirm erscheinen können, sind natürlich Programme notwendig. Diese werden in php/html geschrieben. Für einen Anfänger, der sich damit noch nie beschäftigt hat, zugegebenermaßen eine Hürde. Aber hier gibt es erstens gute Bücher und Schriften und außerdem hat man auch die ganze Internetgemeinde als Helfer bei Fragen und Problemen zur Verfügung. Man glaubt gar nicht, wie viele Antworten zu ganz speziellen Fragen in den einschlägigen Foren bereits gespeichert sind. Man muss nur sehr gezielt über die Suchmaschine die Frage stellen.

3. Der Aufbau der Homepage

Dieses Konzept habe ich in Anlehnung an mehrere im Internet bereits vorhandenen Homepages entwickelt und benutze es selbst für mehrere meiner Internetauftritte und auch für Homepages einiger Freunde. Mein Konzept ist nicht besonders kompliziert, natürlich ist es für Spezialisten auch nicht mehr der letzte Schrei, es hat jedoch Vorteile, z.B. die Anzahl der Coding-Zeilen ist sehr klein, daher sind die Programme sehr übersichtlich und auch für Anfänger zu durchschauen. Ich benutze eine Fenstertechnik, wobei das linke Fenster die Navigationsleiste ist, das Top-Fenster die Überschrift und das rechte große Fenster enthält die eigentlichen Seiten. Nachdem dem Navigationsfenster auch eine SQL-Tabelle zu Grunde liegt, kann man sehr variabel seine Einteilung der aufzurufenden Seiten gestalten. Es gestattet natürlich Untermenüs, die man auch weiter stufen kann. Die Steuerung ist der Aufruf der ersten Seite, eine Datei mit Namen index.html.

Hier nun mein Aufrufprogramm für meine Internetseite Dokumentation: index.html

In der Auflage 2 habe ich php 7.2 verwendet, das bei den mysql-Befehlen andere Parameter hat. Zu den wesentlichen Änderungen von php 5 zu php 7 kommt später ein eigenes Kapitel.

```
<html>
<head>
<style type="text/css">
  @import url(„FORMATE.CSS“) screen;
</style>
<title>Dokumentation Heinz Wember</title>
</head>
<frameset rows="130,*">
<!-- Dadurch ergeben sich zwei Frame-Fenster, deren Inhalt hier bestimmt wird.
  Das erste Frame-Fenster wird normal definiert, anstelle des zweiten
  folgt jedoch das folgende untergeordnete Frameset: -->
  <frame name="top" src="top.php" scrolling="no" frameborder="no">
<frameset cols="15%,85%"rows=1200 >
  <frame name="navigation" src="navigation.php" scrolling="auto" frameborder="no" >
```

```

<frame name="content" src="uebersicht.php" scrolling="auto" frameborder="no">
</frameset>
</frameset>

```

```

<body>
Sie befinden sich auf der Homepage Heinz-Wember.de 05-Jun-2013
</body>
</noframes>
</body>
</html>

```

Wie man sieht, ist die Definition der Fenstergrößen relativ programmiert, also mit % des Bildschirms. Dieses Startprogramm ruft die entsprechenden Programme für die drei definierten Fenster auf: top, navigation und content.

4. Die einzelnen Fenster

4.1 Das Fenster top

kann praktisch alles enthalten, es kann eine html-Datei oder auch eine php-Datei sein, hier bei diesem Beispiel top.php ist es nur eine Überschrift, man kann ebenso gut auch eine Zeichnung oder ein Foto verwenden.

```

<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
<style type="text/css">
  @import url(„FORMATE.CSS“) screen;
</style>
<title>Heinz Wember Dokumentation</title>
<meta name="author" content="Heinz Wember Augsburg, top.php, 05-Jun-2013">
</head>
<body>
<center><big><big><strong>
<br>Heinz Wember Dokumentationen
</strong></big></big></center>
</body>
</html>

```

4.2 Das Fenster navigation

Ist eine php-Datei, die die Datei navigation.php aufruft.

Hier die Felder dieser Tabelle bzw. gleich der Befehl, um diese Tabelle aufzubauen.

```

<!DOCTYPE html PUBLIC „-//W3C//DTD HTML 4.01 Transitional//EN“>
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
<style type="text/css">
  @import url(„FORMATE.CSS“) screen;
</style>

<meta name="author" content="Heinz Wember Augsburg, php7, 16-Feb-2019">

</head>

```



```

<left>
<small>
<table width=15%; text-align: left; margin-left: auto; margin-right: auto; border="1" cellpadding="2" cellspacing="2">
  <tbody>
    <?php
      include „../db/zugriff-dbkriegshaber.inc.php“;
      $abfrage = „SELECT `gruppe` , `ugruppe` , `titel`
        FROM `navigation`
        WHERE `gruppe` = ‚00‘ and aktiv = ‚j‘
        ORDER BY `ugruppe`“;
      $ergebnis = mysqli_query($db, $abfrage);
      while($row = $ergebnis->fetch_assoc())
      {
        echo „<tr> $row[titel] </tr> <br>“;
        $gruppe = $row[gruppe];
      }
      mysqli_close($db);
    ?>
  </tbody>
</table>
</small>
</body>
</html>

```

```

CREATE TABLE `navigation` (
  `id` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `gruppe` char(2) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL DEFAULT ‚00‘,
  `ugruppe` char(2) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `aktiv` varchar(1) NOT NULL DEFAULT ‚j‘,
  `titel` varchar(255) CHARACTER SET latin1 COLLATE latin1_german1_ci DEFAULT NULL,
  `text` text CHARACTER SET latin1 COLLATE latin1_german1_ci,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`),
  KEY `n_gruppe` (`gruppe`)
) ENGINE=MyISAM AUTO_INCREMENT=1522 DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;

```

Die notwendigen Felder sind gruppe, ugruppe, aktiv und titel, die andern sind sinnvoll. Das Feld id sollten eigentlich alle Tabellen haben mit einem Primärindex, das Feld text dient nur zur eigenen Dokumentation, das Feld timestamp erstellt einen Zeitstempel, an Hand dessen man feststellen kann, wann dieser Satz zuletzt geändert bzw. neu erstellt wurde.

4.3 Das Hauptfenster mit Namen content

Hier gleich ein Beispiel für ein einfaches Programm, es ist das Programm, das einen Text einer Tabelle aufruft, siehe hier.

Die dazugehörige Tabelle ist hier.

```

CREATE TABLE `dokumentation` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `text` text COLLATE latin1_german1_ci NOT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM AUTO_INCREMENT=4 DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Datenbankkonzept' AUTO_INCREMENT=1 ;

```

Diese Tabelle hat drei Felder, wobei es nur auf das Feld text ankommt. Dieser Text wird mit dem Programm dokumentation.php auf dem Bildschirm dargestellt. Diese Tabelle enthält nur einen Satz.

Nun ein Beispiel für eine Tabelle, die mehrere Sätze enthält. Es ist die Tabelle uebersicht. Sie wird mit dem Programm uebersicht.php dargestellt. Der Aufbau der Tabelle steht hier.

```

CREATE TABLE `uebersicht` (
  `id` tinyint(4) NOT NULL AUTO_INCREMENT,
  `name` varchar(255) CHARACTER SET latin1 COLLATE latin1_german1_ci DEFAULT NULL,
  `gruppe` varchar(255) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL DEFAULT 'Genealogie',
  `sort` int(5) NOT NULL,
  `active` varchar(1) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL DEFAULT 'j',
  `programm` text CHARACTER SET latin1 COLLATE latin1_german1_ci,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM AUTO_INCREMENT=106 DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;

```

```

<!DOCTYPE html PUBLIC „-//W3C//DTD HTML 4.01 Transitional//EN“>
<html xmlns=“http://www.w3.org/1999/xhtml“ xml:lang=“de“>
<head>
  <style type=“text/css“>
    @import url(„FORMATE.CSS“) screen;
  </style>
  <meta name=“author“ content=“Heinz Wember Augsburg, php7, 09-Mar-2019“>
  <title>Straßen in Kriegshaber</title>
</head>

<body>
<br><center>
<h1>Straßen von Kriegshaber<BR>Alle Sätze von strassen2 ausgeben, verknüpft mit Tabelle strassen1 </h1>
<br>
<?php
  define(„UNLOCK“,1);
  include „../db/zugriff-dbkriegshaber.inc.php“;

  echo „<td style=width: 5%; font-family: Arial; font-weight: bold;>$row[falk2008]</td></tr>“;
}
mysqli_close($db);
?>

```

```

$abfrage = „SELECT strassen2.id, strassen2.strnralt, statamtstrassen.schluessel, statamtstrassen.strname,
strassen2.status1913,
    strassen2.status1938, strassen2.status2008,
    strassen2.bemerkung, strassen2.foto1, strassen2.foto2, strassen2.foto3, strassen2.verknuepfung1,
strassen2.verknuepfung2, strassen2.falk2001, strassen2.falk2008
FROM statamtstrassen , strassen2
WHERE strassen2.schluessel = statamtstrassen.schluessel
ORDER BY statamtstrassen.strname“;

```

```

$ergebnis = mysqli_query($db, $abfrage);

```

```

?>
<table style=“text-align: left; width: 100%;“ border=“0“ cellpadding=“2“ cellspacing=“2“>

```

```

<tbody>
<tr>
<td style=width: 2%;>ID</td>
<td style=width: 3%;>Schlüssel</td> <td style=width: 15%;>Name</td>
<td style=width: 5%;>Status1913</td>
<td style=width: 5%;>Status1938</td>
<td style=width: 5%;>Status2008</td>
<td style=width: 30%;>Bemerkung</td>
<td style=width: 5%;>Foto1</td>
<td style=width: 5%;>Foto2</td>
<td style=width: 5%;>Foto3</td>
<td style=width: 5%;>Verkn&uuml;pfung1</td>
<td style=width: 5%;>Verkn&uuml;pfung2</td>
<td style=width: 5%;>Falk2001</td>
<td style=width: 5%;>Falk2008</td>
</tr>

```

```

<?php

```

```

while($row = $ergebnis->fetch_assoc())

```

```

{
    echo „<tr><td style=width: 2%; font-family: Arial; font-weight: bold;>$row[id]</td>“;
    echo „<td style=width: 15%; font-family: Arial; font-weight: bold;>$row[schluessel]</td>“;
    echo „<td style=width: 15%; font-family: Arial; font-weight: bold;>$row[strname]</td>“;
    echo „<td style=width: 5%; font-family: Arial; font-weight: bold;>$row[status1913]</td>“;
    echo „<td style=width: 5%; font-family: Arial; font-weight: bold;>$row[status1938]</td>“;
    echo „<td style=width: 5%; font-family: Arial; font-weight: bold;>$row[status2008]</td>“;
    echo „<td style=width: 30%; font-family: Arial; font-weight: bold;>$row[bemerkung]</td>“;
    echo „<td style=width: 5%; font-family: Arial; font-weight: bold;>$row[foto1]</td>“;
    echo „<td style=width: 5%; font-family: Arial; font-weight: bold;>$row[foto2]</td>“;
    echo „<td style=width: 5%; font-family: Arial; font-weight: bold;>$row[foto3]</td>“;
    echo „<td style=width: 5%; font-family: Arial; font-weight: bold;>$row[verknuepfung1]</td>“;
    echo „<td style=width: 5%; font-family: Arial; font-weight: bold;>$row[verknuepfung2]</td>“;
    echo „<td style=width: 5%; font-family: Arial; font-weight: bold;>$row[falk2001]</td>“;

```

```

</tbody>

```

```

</table>

```

```

<hr>

```

```

<a target=“_blank“ href=“index-kat.html“>Übersicht Kataster</a>

```

```

</center>

```

```

<hr>

```

```

<small>Änderungsstand: 18-Feb-2019<br>

```

```
<a href=“../Impressum.htm“>Heinz Wember</a></small>  
</body>  
</html>
```

Systematik der Tabelle navigation

Hier nun die erste Seite der Tabelle.

Connect to MySQL-DB

00	00	j	Hauptmenü	Hauptmenü
00	05	j	Strassennamen und Hausnummern	Straßennamen und Hausnummern
00	06	j	Strassen nach Straßenschlüssel sortiert	Strassen nach Straßenschlüssel sortiert
00	07	j	Geschichte der Hausnummern	Geschichte der Hausnummern
00	08	j	Strassen mit Strassenschlüssel sortiert nach Namen	Strassen mit Straßenschlüssel sortiert nach Namen
00	10	j	Hausnr alt zu Hausnr heutiger Adressierung	Hausnr alt zu Hausnr heutiger Adressierung
00	11	j	Hausnr alt zu Hausnr heutiger Adressierung, nur Grundstücke mit Adresse	Hausnr alt zu Hausnr heutiger Adressierung, nur Grundstücke mit Adresse
00	14	j	Hausnr alt zu Hausnr neu nach altem Strassennamen sortiert	Hausnr alt zu Hausnr neu nach altem Straßennamen sortiert
00	16	j	Hausnr alt zu Adressierung von 1917	Hausnr alt zu Adressierung von 1917
00	17	j	Hausnr neu zu Hausnr alt, 1917	Hausnr neu zu Hausnr alt 1917
00	18	j	Crossreference nach Eigentümer	Crossreference nach Eigentümer
00	19	j	alle Häuser heute	alle Häuser heute
00	20	j	Auswertungen fürs Buch	Auswertungen fürs Buch
00	21	j	alle Häuser heute	alle Häuser heute
00	25	j	Details zu den Straßen von Kriegshaber	Details zu den Straßen von Kriegshaber
00	30	j	Kataster Auswertungen	Kataster-Auswertungen
00	34	j	Ablösung der Grundlasten	Ablösung der Grundlasten
00	40	j	Kirchenbücher	Kirchenbücher
00	50	j	Geschichte Immobilien der Juden	Gesch Immobilien der Juden
00	52	j	Geschichte einiger Häuser, u.a. Kirche, Synagoge	Geschichte einiger Häuser, u.a. Kirche, Synagoge
00	54	j	Fotos Kriegshaber	Fotos Kriegshaber
00	57	j	Strassennamen in Kriegshaber	Straßennamen in Kriegshaber
00	58	j	Grenzen von Kriegshaber seit 1810	Grenzen von Kriegshaber seit 1810
00	59	J	Status 1920	Status 1920
00	60	j	Strassennamen 1938/heute	Straßennamen 1938/heute
00	61	j	Strassen in Augsburg nach Namen	Straßen in Augsburg nach Namen
00	62	j	Strassen in Augsburg nach Bezirknr	Straßen in Augsburg nach Bezirknr
00	64	j	Quellen für Kriegshaber	Quellen für Kriegshaber
00	65	j	Quellen StAA	Quellen StAA
00	66	j	Quellen Bistums-AA	Quellen Bistums-AA
00	67	j	Quellen StadtAA	Quellen StadtAA
00	68	j	Quellen Fotos/Bilder Übersicht	Quellen Fotos/Bilder Übersicht
00	69	j	Quellen Fotos/Bilder StadtAA	Quellen Fotos/Bilder StadtAA
00	70	j	Quellen Heimatforschung	Quellen Heimatforschung
00	71	j	Seminar Kriegshaber	Seminar Kriegshaber
00	71	j	sonstige Quellen	sonstige Quellen

Beispiele für Select-Befehle

Einfacher Select: eine Tabelle sortiert ausgeben:

```
SELECT mann, frau, jahr  
FROM hochzeiten  
ORDER BY frau, jahr
```

Select, um eine Tabelle mit nur einem Satz anzuzeigen:

```
SELECT text  
FROM dokukatasterdaten
```

Die Sätze einer Tabelle zählen und ausgeben nach Gruppen pro Jahr:

```
SELECT jahr, count(*) AS summejahr  
FROM hochzeiten  
GROUP BY jahr  
ORDER BY jahr
```

Die Straßennamen sortiert ausgeben:

```
SELECT strassenname, stadtbnr, stadtbezirk  
FROM strassenstadt  
ORDER BY strassenname
```

Alle Sätze der Tabelle Hochzeiten zählen und pro Beruf der Frau summiert ausgeben

```
SELECT fberuf, count(*) AS summeberuf  
FROM hochzeiten  
GROUP BY fberuf  
ORDER BY fberuf
```

Die Vorname der männlichen Hochzeiter ausgeben:

```
SELECT mann, mvorname  
FROM hochzeiten  
ORDER BY mvorname
```

Befehl für das Navigations-Programm:

```
SELECT `gruppe`, `ugruppe`, `titel`  
FROM `navigation`  
WHERE `gruppe` = '00'  
AND aktiv = 'j'  
ORDER BY `ugruppe`
```

Select um das Untermenü 30 aufzurufen:

```
SELECT `gruppe`, `ugruppe`, `titel`  
FROM navigation  
WHERE `gruppe` = '30'  
AND aktiv = 'j'  
GROUP BY `ugruppe`
```

Programm, um alle Häuser zu zeigen mit der verknüpften Tabelle Straßennamen:

```
SELECT haeuser.hausnr1913, haeuser.plannr, haeuser.schluessel1913, haeuser.schluessel1918,  
haeuser.strassennr, statamtstrassen.strname AS strassenname2012, haeuser.hausnrneu, haeuser.besitzer1913  
FROM haeuser  
LEFT JOIN statamtstrassen ON haeuser.strassennr = statamtstrassen.schluessel ORDER BY hausnr1913
```

Befehl, um alle Katasterbände anzuzeigen

```
SELECT katbandn.id AS id, katkopf.raort AS raort, katkopf.rentamt AS ra,
       katkopf.steuergemeinde AS stg, katkopf.steuerdistrikt AS std,
       katkopf.landamtsger AS ger,
       katabk.name, katabk.abk AS abk,
       katbandn.typ AS typ, kattyprn.gruppe AS gruppe,
       katbandn.bandnr AS bnr, katbandn.ksv AS ksv, katbandn.ksb AS ksb,
       katbandn.hv AS hv, katbandn.hb AS hb, kattyprn.jahr AS jahr, kattyprn.jahr AS jahr
FROM katbandn
LEFT JOIN katkopf ON katkopf.raort = katbandn.raort
LEFT JOIN kattyprn ON katbandn.typ = kattyprn.typ
LEFT JOIN katabk ON kattyprn.abk = katabk.abk
ORDER BY katkopf.steuergemeinde, kattyprn.gruppe, kattyprn.typ, katbandn.ksv, katbandn.bandnr
```

Anzeigen von Personen mit diversen Daten (u.a. Alter), die Variablen \$fam, \$jahr1 und \$jahr2 werden vorher eingegeben

```
SELECT geburtdetail.fam AS fam, knachname, kvorname, kgebdatum, DATE_Format(kgebdatum, '%Y')
AS kgebjahr, quelle, ktoddatum,
(YEAR(ktoddatum) - YEAR(kgebdatum)) AS kalterj,
(MONTH(ktoddatum) - MONTH(kgebdatum)) AS kalterm,
(DAY(ktoddatum) - DAY(kgebdatum)) AS kaltert,
vnachname, vvorname, vgebjahr, DATE_Format(kgebdatum, '%Y') - vgebjahr AS valter,
mnachname, mvorname, mgebjahr, DATE_Format(kgebdatum, '%Y') - mgebjahr AS malter
FROM `geburtdetail`
WHERE ktoddatum IS NOT NULL
AND kgebdatum IS NOT NULL
AND (YEAR(ktoddatum) - YEAR(kgebdatum)) < 1
AND fam = '$fam'
AND DATE_Format(kgebdatum, '%Y') >= $jahr1
AND DATE_Format(kgebdatum, '%Y') <= $jahr2
OR ktoddatum IS NOT NULL
AND kgebdatum IS NOT NULL
AND (YEAR(ktoddatum) - YEAR(kgebdatum)) < 2
AND (MONTH(ktoddatum) - MONTH(kgebdatum)) < 0
AND fam = '$fam'
AND DATE_Format(kgebdatum, '%Y') >= $jahr1
AND DATE_Format(kgebdatum, '%Y') <= $jahr2
ORDER BY knachname, kvorname
```

Anzeigen von Personen, die nicht ein Jahr geworden sind

```
SELECT geburtdetail.fam AS fam, knachname, kvorname, kgebdatum, DATE_Format(kgebdatum, '%Y')
AS kgebjahr, quelle, ktoddatum,
(YEAR(ktoddatum) - YEAR(kgebdatum)) AS kalterj,
(MONTH(ktoddatum) - MONTH(kgebdatum)) AS kalterm,
(DAY(ktoddatum) - DAY(kgebdatum)) AS kaltert,
vnachname, vvorname, vgebjahr, DATE_Format(kgebdatum, '%Y') - vgebjahr AS valter,
mnachname, mvorname, mgebjahr, DATE_Format(kgebdatum, '%Y') - mgebjahr AS malter
FROM `geburtdetail`
WHERE ktoddatum IS NOT NULL
AND kgebdatum IS NOT NULL
AND (YEAR(ktoddatum) - YEAR(kgebdatum)) < 1
OR ktoddatum IS NOT NULL
```

```
AND kgebdatum IS NOT NULL
AND (YEAR(ktoddatum) - YEAR(kgebdatum)) < 2
AND (MONTH(ktoddatum) - MONTH(kgebdatum)) < 0
ORDER BY knachname, kvorname
```

Anzeigen von Personen, die leben und heute über 20 Jahre sind und auch diejenigen, die in einem Alter über 20 Jahre verstorben sind

```
SELECT geburtdetail.fam AS fam, knachname, kvorname, kgebdatum, DATE_Format(kgebdatum, '%Y')
AS kgebjahr, quelle, ktoddatum,
(YEAR(ktoddatum) - YEAR(kgebdatum)) AS kalterj,
(MONTH(ktoddatum) - MONTH(kgebdatum)) AS kalterm,
(DAY(ktoddatum) - DAY(kgebdatum)) AS kaltert,
vnachname, vvorname, vgebjahr, DATE_Format(kgebdatum, '%Y') - vgebjahr AS valter,
mnachname, mvorname, mgebjahr, DATE_Format(kgebdatum, '%Y') - mgebjahr AS malter
FROM `geburtdetail`
WHERE ktoddatum IS NULL
AND kgebdatum IS NOT NULL
AND (DATE_FORMAT(CURDATE(), '%Y') - DATE_FORMAT(kgebdatum, '%Y')) > 20
OR ktoddatum IS NOT NULL
AND kgebdatum IS NOT NULL
AND (YEAR(ktoddatum) - YEAR(kgebdatum)) > 20
ORDER BY knachname, kvorname
```

Doppelte Sätze rausfiltern:

```
DROP TABLE If EXISTS tmp;
CREATE TABLE tmp SELECT plannr, hausnr1913, hausnrneu1918, hausnrheute,
count(*) as anzahl
FROM haeuser
group by $feld;
SELECT plannr, hausnr1913, hausnrneu1918, hausnrheute, anzahl
FROM tmp
WHERE anzahl > 1;
```

Beispiel, wie ich untersuche, ob es doppelte Steuerzeichen (gruppe und ugruppe) in der Datei navigation gibt:

```
DROP TABLE If EXISTS tmp
CREATE TABLE tmp SELECT gruppe, ugruppe, count(*) as anzahl
FROM navigation
SELECT gruppe, ugruppe, anzahl
FROM tmp
WHERE anzahl > 1
?>
```


Beispiel für relative und feste Spaltenabstände

Als Beispiel wurde hier ein Demosystem verwendet, mit dem man seine Fotos dokumentieren kann.

Es werden alle erfassten Fotos (eingescannte Dias, Negative und Digitalfotos) gezeigt. Es werden die Daten in zwei Tabellen erfasst:

Tabelle gruppe: hier wird für die Gruppe eine Abkürzung und die dazugehörige Beschreibung erfasst.

Tabelle fotos: hier werden die Felder Fotonummer, Abkürzung für Gruppenzugehörigkeit, Jahr der Aufnahme, Personenbeschreibung und Motivbeschreibung erfasst und evtl. der Link auf das Foto, falls es bereits im Internet steht, oder ein Minibild wird direkt gezeigt.

Alle Abkürzungen für die Gruppenzugehörigkeit müssen auch in der Tabelle gruppe erfasst sein. Dies wurde so gemacht, damit man bei der Erfassung der Fotos nicht so viele Buchstaben dafür erfassen muss.

Für die Erfassung der Bildnummer gibt es mehrere Möglichkeiten, für die man sich vorher entscheiden muss:

- Die Nummer ist einzig, d.h. es darf diese Nummer nicht noch einmal vorkommen, eine Möglichkeit dafür wäre z.B. jjjjmmnnnn, also Jahr, Monat und eine beliebige Nummer

- Man verwendet die im Feld Jahr eingegebene Jahreszahl auch für die Nummerierung, dann ist natürlich die Fotonummer kürzer, also mmnnnn

- Wenn schon jetzt die Nummerierung innerhalb einer Gruppe erfolgt (in jeder Gruppe kann die laufende Nummer 1 vorkommen), dann kann auch die Abkürzung zur Nummerierung verwendet werden, also mmnnnn, wobei die Nummer mmnnn innerhalb der Gruppe und Jahr einzig sein muss, aber die Fotonummer kann in jedem Jahr und jeder Gruppe jeweils mit 1 beginnen. Dann ist es so, dass die Bildnummer aus Abkürzung, Jahr und Nummer besteht.

Wenn das eingegebene Jahr zur Identifizierung des Bildes verwendet wird, ist es unbedingt einzugeben, wenn das Jahr unbekannt ist, notfalls das Jahr mit 0000 oder sonst einem vernünftigen numerischen Wert versehen. Das eingegebene Jahr wird ja zur Sortierung verwendet, ebenso wie die Abkürzung. Also die Abkürzung ist ebenso ein Pflichtfeld, notfalls hat es den Wert "unbekannt" oder "noch zu klären".

Zur Erfassung des Fotos bzw. des Minifotos:

Der Link auf ein Foto, das bereits im Internet steht, ist in meinem Beispiel: `07091322`

der Wert im Beispiel des Minifoto ist folgendes: ``

Ich habe meine Fotonummern nach Jahr-Monat oder auch Jahr-Monat-Tag mit angehängter laufender Nummer aufgebaut. Nachdem die Fotos in Verzeichnissen stehen, die als Oberverzeichnis Jahr und als Unterverzeichnis Monat haben, kann ich mir bei der Dokumentation die Erfassung des Verzeichnisses sparen.

Zur Auswahl

Das Zeichen % ist eine Wildcard wie bei DOS/Windows *. Wenn also %Hans% eingegeben wird, werden alle Sätze gefunden, bei denen im Feld personen Hans auftaucht, egal ob am Anfang, in der Mitte oder am Ende. Hans% wäre auch eine Suchmöglichkeit, jedoch muss dann Hans am Anfang stehen. Hans ist auch eine Suchmöglichkeit, hierbei werden nur Sätze angezeigt, in denen nur das Wort Hans im Feld personen steht.

Achtung: G und g können verschiedene Zeichen sein. Wenn bei %Grat% nichts gefunden wird, könnte aber bei %grat% durchaus etwas gefunden werden. Evtl. nach %rat% suchen.

Dies ist zwar bei meiner Datenbank bei 1und1 nicht der Fall, ich bin aber nicht sicher, ob es bei anderen Installationen genauso ist.

Zur Programmierung

Ich habe der Einfachheit halber für beide Versionen dieselbe Datenbank genommen, sowohl Definition wie auch Daten. So kann man das unterschiedliche Verhalten bei der Anzeige bei Version 1 und Version 2 gut verfolgen.

Ich habe zwei verschiedene Programmiermöglichkeiten gewählt, die sich nur in dem einen Punkt unterscheiden, bei Version 1 steht die Überschrift zu den Listen nur unmittelbar über den Daten und bei Version 2 liegt die Überschrift außerdem noch im darüber liegenden Fenster (top genannt).

Zur Erstellung der html-Datei, die aus den php-Einschüben generiert wird, ist folgendes Interessante zu sagen: Man kann die Feldbreite relativ und absolut definieren, ich persönlich bevorzuge die relative Definition, also Feld z.B. 20% der gesamten Fensterbreite, absolut wäre z.B. 200 Pixel. Wenn aber das php-Programm interpretiert wird, dann nimmt der Interpreter Rücksicht auf die tatsächliche Zeichenanzahl pro Feld. D.h. je nach Länge der Daten kann die Feldbreite variieren. Das ist natürlich in der Regel wünschenswert, weil so das Aussehen auch besser ist als wenn es starr wäre. Die Kehrseite dieser Medaille ist aber, dass dies nur dadurch erreicht werden kann, wenn man die Überschrift in ein anderes Fenster stellt und die Feld- und Fensterbreiten fest definiert, also in Pixel. Daher habe ich die Programme in Version 2 index.html und top.html mit Pixel definiert. Beim Programm fotosnachablageort habe ich ebenfalls die fixe Definition gewählt. Hier muss man sich entscheiden, welche Bildschirmbreite man verwenden will. Nachdem bekannt ist, dass die meisten Benutzer eine Bildschirmbreite von 1024 Pixel haben, habe ich bei diesem Programm eine Breite von 1000 Pixel gewählt. Wie man sieht, steht die Überschrift im top-Fenster genau über den Daten. Dies bleibt auch so, wenn man die Fensterbreite von 1000 Pixel nach 1600 Pixel verändert. Geht man jedoch unter 1000 Pixel Fensterbreite verschwindet ein Teil des Datenfensters. Man kann jedoch noch die Fensterbreite der Daten zu Lasten des Navigationsfensters erweitern (mit der Maus auf die Trennlinie zwischen linkem und rechtem Fenster gehen. Wenn sich der Cursor verändert, kann man mit der gedrückten linken Maustaste dies kontinuierlich durchführen), aber mit 800 Pixel Fensterbreite ist alles ausgereizt.

Nun zum Unterschied die relative Programmierung beim Programm fotonachgruppeundjahr.php. Hier kann das Programmfenster wesentlich kleiner gewählt werden und man kann immer noch alle Daten lesen, ohne die Lauffleiste zu bemühen. Jetzt ist die Spaltenbreite nur durch die Breite des Fotos in Spalte Foto (150 Pixel) bzw. durch das längste Wort in den anderen Spalten begrenzt, also etwa 16 Zeichen.

Noch ein Hinweis: ich habe bewusst auf den Programmen für das Datenfenster (content genannt) immer einen Rücksprung definiert. Die Suchmaschinen lösen die einzelnen Programme des Gesamtfensters auf, so dass es vorkommt, dass nach Ausführen eines Links aus der Suchmaschine nur das Fenster content gezeigt wird. Nun sind die anderen Fenster (navigation und top) nicht mehr zu sehen. Jedoch ist der Rücksprung so programmiert, dass die ursprüngliche Fensterkonzeption wieder aufgebaut wird (Übersicht).

N. B. Bei dieser gezeigten Applikation ist es ohnehin nicht wichtig, die Feldüberschrift immer im Blickfeld zu haben, weil der Dateninhalt selbst erklärend ist.

Diese beiden Foto-Dokusysteme stehen mit der dazugehörigen Dokumentation auf meiner Homepage.

Fotodoku1: <https://heinz-wember.de/MySQL-rel-FotoDoku/index.html>

Fotodoku2: <https://heinz-wember.de/MySQL-fix-FotoDoku/index.html>

-- Anhang zu Buch Die Datenbank MySQL als Hilfsmittel für die -- Darstellung geschichtlicher Themen: Datei
gestaltung.txt

Textgestaltung mit html

Dies ist eine Zeile zwischen h1

Dies ist eine Zeile zwischen h2

Dies ist eine Zeile zwischen h3

Dies ist eine Zeile zwischen h4

Dies ist eine Zeile zwischen h5

dies ist ein Text zwischen bold

dies ist ein Text zwischen strong

alle html-Steuerzeichen sind mit spitzer Klammer auf und spitzer Klammer zu gekennzeichnet. Die Zeichen spitze Klammer auf und zu dürfen also nicht als Textzeichen verwendet werden. Die Zeichen spitze Klammer stehen auf der Tastatur links neben y. Der Text wird also zwischen diesen Steuerzeichen eingeschlossen:

im Beispiel bold: es kommt spitze Klammer auf dann das Zeichen b dann spitze Klammer zu; nun folgt der Text, der entsprechend hervorgehoben ist, dann das Endezeichen, das wie das Anfangzeichen ist, jedoch zuvor einen Schrägstrich hat, also spitze Klammer auf Schrägstrich und dann das eigentliche Steuerzeichen (in diesem Beispiel b) und spitze Klammer zu.

Genauerer siehe im [Handbuch](#)

Wie nun die einzelnen html-Steuerzeichen interpretiert werden, ist in der Formatdatei (bei mir FORMAT.CSS) definiert, die im Root-Verzeichnis steht. Man kann natürlich für unterschiedliche Unterverzeichnisse auch unterschiedliche Formatdateien nehmen, was ich aber nicht mache, weil ich auf ein einheitliches Erscheinungsbild achte.

[Datei FORMATE.CSS](#)

Diese Homepage ist so designed, dass bei allen Fenstergrößen, egal ob 800 x 600, 1024 x 768 oder auch 1680 x 1050 Pixel die Bildschirmgröße optimal genutzt wird. Bei kleinen Fenstergrößen erscheint das Navigationsfenster natürlich nicht vollständig auf dem Bildschirm. Dann kommt die Windows-Laufleiste. Außerdem kann man die Breite des Navigationsfenster auf Kosten des großen rechten Fensters mittels linker Maustaste am Fensterrand vergrößern.

Im Großen und Ganzen sind die gängigen Browser in der Interpretation der Steuerzeichen alle gleich. Die Farben der

Gestaltung

Schriften (bei h1 bis h5 und bold) sind nur sichtbar bei den Browsern Mozilla Firefox, Opera und Google Crom, nicht jedoch bei Microsoft Internet Explorer 8.0 in Verbindung Windows Vista. Bei Google Chrom ist mir aufgefallen, dass das Verändern der Breite des Fensters Navigation nicht funktioniert.

Änderungsstand gestaltung: 28-Jan-2014 Upd 25-Mar-2020

file:///F:/Wember/Heinz/BuchMySQL/Auswertungen/tmp/Gestaltung.html[26.03.2020 04:47:40]

Ein Beispiel für eine interaktive Abfrage

Diese Abfrage soll beantworten, wo überall ein bestimmter alphanumerischer Wert vorkommt, das kann sein ein Name, ein Ort, eine Flurnummer (z.B. P#437/002), eine Katasterseite (z.B. KS#300), ein Datum, etc. Hierbei muss man erst festlegen, wo solche Begriffe vorhanden sein können. Auf jeden Fall müssen es Felder mit alphanumerischen Werten sein, alle numerischen Felder scheiden also aus. Dann ist vorher festzulegen, wie diese Felder strukturiert sind. Bei Namen ist es klar, z.B. Bauer oder Michael, wie ist es mit Flurnummern, Katasterseiten und Datum? Wenn man diese Werte so in die Tabelle einträgt, wie sie in der Quelle stehen, hat man keine Chance, ein einwandfreies Ergebnis zu bekommen. Erstens hieß es bis ca. 1937 Plannummer, später Flurnummer, dann hat sich die Struktur geändert, früher hießen die abgetrennten Teilfluren 437 1/2, 437 1/11, 437 1/111. Später stand in der Quelle für dieselben Flurstücke 437/2, 437/11 und 437/111. Also mit dieser Systematik ist es wieder unbrauchbar. Ich habe die Flurnummern, damit sie auswertbar sind, systematisiert: also 437/002, 437/011 und 437/111. Bei kleineren Grundstücken, die maximal nur 9 Teilfluren haben, genügt 001/1, 001/2, 001/3 etc. D.h. ich habe auch die Flurnummern unter 100 auf drei Stellen erweitert. Erst jetzt macht es Sinn, eine Auswahl zu treffen.

Nun zu den Werten, die man eingeben kann. Enthält das Feld, nach dem gesucht wird, nur ein Wort, z.B. Müller, dann kann der Suchbegriff Müller sein, damit dieser Satz auch gefunden wird. Heißt der Wert in dem Feld statt Müller z.B. Müllerfreund, so wird der Satz mit Auswahl Müller nicht gefunden. Nun ist es aber so, dass fast alle alphanumerischen Felder sehr viele Worte enthalten, der Wert Müller kann am Anfang oder in der Mitte oder am Ende stehen. Damit man alle drei Varianten abdecken kann, sollte man %Müller% abfragen, dann kann Müller am Anfang, in der Mitte oder am Ende stehen, der Satz wird dann angezeigt.

Wie ist es mit mehreren Begriffen? Grundsätzlich ist eine Abfrage immer eine UND-Verknüpfung, also %Müller%Anton% wird immer gefunden, wenn im Satz sowohl Müller als auch Anton steht, und zwar in dieser Reihenfolge. Steht zuerst Anton im Feld und dann Müller wird dieser Satz nicht gefunden.

Wie ist es mit ODER-Verknüpfungen? Ich will alle Sätze finden, wo entweder Meier, Maier, Meyer, Mayer etc. vorhanden ist. Hier muss man in diesem Fall vier Abfragen starten, jedes mal einen Begriff z.B.%Meier% nehmen.

Das Suchprogramm heißt haeusersuchnamen.php

```
<!DOCTYPE html PUBLIC „-//W3C//DTD HTML 4.01 Transitional//EN“>
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url(„FORMATE.CSS“) screen;
  </style>
  <meta name="author" content="Heinz Wember Augsburg, php7, 19-Feb-2019">
  <title>Stra&szlig;en in Kriegshaber</title>
</head>
<body>
<br><center>
<h1> Suchen nach Begriffen (Namen, Berufe etc) in den Feldern für Eigentümer, sortiert Hausnummer
1913</h1><br>
</center><left>
<br>
<?php
include „../db/zugriff-dbkriegshaber.inc.php“;
echo „<span class='feld2'><input type='radio' name='name' value='bitte eingeben'>“;
if (!$_POST[„gesendet“] && !$_POST[„name“])
```

```

{
?>

<form method="POST" action="<?php echo $PHP_SELF ?>">
<p>Name z.B. %Steppich% oder %Jakob Beisele% oder %Schuhmachermeister% oder %073a%:
  <input type="text" name="name" size="30">
</p>
  <input type="submit" name="Submit" value="Submit">
  <input type="hidden" name="gesendet" value="1">
</form>

<?php
  }

else {

  $name=$_POST[name];

  $abfrage = „SELECT haeuser.hausnr1913, haeuser.plannr, haeuser.strassennr1913,
    haeuser.strassennr1918, statamtstrassen.strname AS strassenname1918,
    statamtstrassen.schluessel , haeuser.hausnrheute, haeuser.besitzer1913, haeuser.hausnr1918,
    haeuser.hausnr1892, haeuser.besitzer1892, haeuser.hausnr1867, haeuser.besitzer1867, haeuser.
hausnr1840,
    haeuser.besitzer1840,
    haeuser.hausnr1810, haeuser.besitzer1810
  FROM haeuser
  LEFT JOIN statamtstrassen ON haeuser.strassennr1918 = statamtstrassen.schluessel
  Where haeuser.besitzer1913 LIKE ‚$name‘
  OR haeuser.besitzer1892 LIKE ‚$name‘
  OR haeuser.besitzer1867 LIKE ‚$name‘
  OR haeuser.besitzer1854 LIKE ‚$name‘
  OR haeuser.besitzer1840 LIKE ‚$name‘
  OR haeuser.plannr LIKE ‚$name‘
  ORDER BY hausnr1913, plannr“;

  echo „Auswahl für $name“;

  $ergebnis = mysqli_query($db, $abfrage);
?>

<table style="width: 85%; text-align: left; margin-left: left; margin-right: auto;" border="1" cellpadding="2"
cellspacing="2">
  <caption></caption><tbody>
    <tr>
      <td style="width: 3%;">HausNr<br> 1913</td>
      <td style="width: 3%;">Plannr</td>
      <td style="width: 3%;">Schl 1913</td>
      <td style="width: 3%;">Schl 1918</td>
      <td style="width: 10%;">StrassenN1918 Hausnr neu</td>
      <td style="width: 20%;">Eigentümer/Bewohner 1913/1917</td>

```

```

<td style="width: 20%;">HNr<br>1892</td>
<td style="width: 20%;">Eigentümer 1892</td>
<td style="width: 20%;">HNr<br>1867</td>
<td style="width: 20%;">Eigentümer 1867</td>
<td style="width: 20%;">HNr<br>1892</td>
<td style="width: 30%;">Eigentümer 1840</td>
<td style="width: 20%;">HNr<br>1810</td>
<td style="width: 30%;">Eigentümer 1810</td>
</tr>

```

```

<?php
while($row = $ergebnis->fetch_assoc())
{
    echo „<tr><td style=width: 6%;>$row[hausnr1913]</td>“;
    echo „<td style=width; 3%;>$row[plannr]</td>“;
    echo „<td style=width; 3%;>$row[strassennr1913]</td>“;
    echo „<td style=width; 3%;>$row[strassennr1918]</td>“;
    echo „<td style=width: 10%;>$row[strassenname1918] $row[hausnr1918]</td>“;
    echo „<td style=width: 20%;>$row[besitzer1913]</td>“;
    echo „<td style=width: 20%;>$row[hausnr1892]</td>“;
    echo „<td style=width: 20%;>$row[besitzer1892]</td>“;
    echo „<td style=width: 20%;>$row[hausnr1867]</td>“;
    echo „<td style=width: 20%;>$row[besitzer1867]</td>“;
    echo „<td style=width: 20%;>$row[hausnr1840]</td>“;
    echo „<td style=width: 20%;>$row[besitzer1840]</td>“;
    echo „<td style=width: 30%;>$row[hausnr1810]</td>“;
    echo „<td style=width: 20%;>$row[besitzer1810]</td></tr>“;
}
mysqli_close($db);
}
?>

```

```

</tbody>
</table>
<hr>
<center>
<a target="_blank" href="index-kat.html">Übersicht Kataster</a>
</center>
<hr>
<small>Änderungsstand Programm: 19-Feb-2019<br>
<a href="../Impressum.html">Heinz Wember</a></small>
</body>
</html>

```

Dokumentation für die Erstellung der Seiten

Vorbemerkung: Ich bringe hier mehrere Beispiele, die aus unterschiedlichen Applikationen stammen. Es gibt systematisch gesehen, mehrere Seiten-Typen.

1. Diejenigen Seiten, die in den jeweiligen Tabellen stehen (z.B. reine Texttabellen, einfache Tabellen und verknüpfte Tabellen, also Texte, die aus einer Verknüpfung von mehreren Tabellen generiert werden) und
2. Seiten, die gar nicht aus einer MySQL-Datenbank generiert werden.

Zu 1:

Hier ist es angebracht, einfache Strukturfelder zu verwenden. Unter einfach verstehe ich, dass die normalen html-Steuerzeichen verwendet werden. Die Steuerzeichen, die von den Programmen von Office (Word, Excel, Powerpoint, Access) erstellt werden, haben eine ganz andere Struktur und vertragen sich im Allgemeinen mit den Standard-html-Steuerzeichen nicht. Wenn der Text in den Tabellen so einfach ist, wie z.B. eine Dokumentationsseite, empfiehlt es sich, den Text ohne irgendwelche Steuerzeichen entweder direkt in die Tabelle zu schreiben oder aber den Text ohne Formatierungszeichen in die Tabelle zu kopieren. Zur "Verschönerung" lassen sich die einfachen html-Steuerzeichen `
` für neue Zeile, `<big>Text</big>` für ein Grad größere Schrift, bzw. `<small>Text</small>` ein Grad kleinere Schrift und `Text` für fette Schrift verwenden.

Will man jedoch ein etwas anspruchsvolleres Bild des Schriftstückes erreichen, dann ist es einfacher, diesen Text mit Hilfe des Editors NVU zu schreiben. Hier kann man dann Tabellen einfügen, eine nummerierte bzw. unsortierte Liste einbauen, etc.

Zu 2:

Hier hat man die Möglichkeit, den Text mit beliebigen Schreibprogrammen zu erstellen und dann entweder das Ergebnis als pdf-Datei auszugeben oder auch aus Word eine html-Datei zu erstellen. Die erstere Methode hat den Vorteil, dass die Texte von den Benutzern nicht leicht geändert werden können, die zweite Methode hat den Vorteil, dass der Aufruf der Seite in der Regel schneller ist und direkt in die Homepage eingebunden wird. Zur Methode Erstellung von html-Seiten aus Word etc. muss gesagt werden, dass hier das endgültige Ergebnis um einiges schlechter ist, als bei der Erstellung desselben Textes mit NVU. Insbesondere die Aufrufzeit der Seite ist bei weitem langsamer, was aber bei kurzen Texten weniger ins Gewicht fällt. Auch ist die Formatierung von fertigen html-Seite von Word nicht mit der Formatierung einer Seite aus NVU kompatibel, was die Eigenschaften der Überschriften, der Hintergrundfarbe, etc. betrifft. Natürlich lässt sich dies nachträglich verbessern, aber dies ist nicht so einfach.

Eine komplexere Seite erstellen mit Hilfe des Programms NVU.

1. Den Text übernehmen, z.B. die betreffende Textpassage von einer bestehenden pdf-Datei mit Adobe Reader anschauen, den Text, den man übernehmen will markieren, `<rechte Maustaste>` `<kopieren>` NVU starten mit leerer Seite, dann den Text hier einfügen `<rechte Maustaste>` `<Einfügen>`
2. Nun steht der Text so in NVU da, wie er im Original zu sehen war, d.h. wenn es sich um kurze Zeilen gehandelt hat, dann sind nun auch hier diese kurzen Zeilen zu sehen.
3. Nun den Text so bearbeiten, dass der Text fortlaufend steht, bei Zeilenende beim NVU beachten, dass keine Trennungsstriche vorkommen, d.h. das komplette Wort steht dann auf der neuen Zeile.
4. Nun evtl. Bilder erstellen, entweder Fotos von einer Digitalkamera, Fotos aus der PDF Datei oder sonstwo her.
 - 4.1. Fotos von der Digitalkamera: das Foto in ein Bildbearbeitungsprogramm laden (ich verwende das Programm Corel Paint Shop Pro oder Photoshop), nun das ganze Bild oder einen Ausschnitt erstellen und die Größe des Bildes so verkleinern, dass es gut in die zu erstellende Seite passt (Bilder mit der Größe von ca. 800 x 600 nur in Ausnahmefällen verwenden, da dieses Bild bei den meisten Anwendern nicht ganz auf den Computerbildschirm passt, besonders dann, wenn die zu erstellende Seite in einem Fenster angezeigt wird, also maximal ca. 400 x 300 Pixel. Dieses erstellte Bild im Format jpg lokal speichern.
 - 4.2. Fotos von einer pdf-Datei verwenden: das Bild im Programm Adobe Reader markieren und kopieren, dieses Bild im Bildbearbeitungsprogramm einfügen (bei Paint Shop Pro: `<Datei>` `<Neu>` die Bildabmessung in

etwa schätzen z.B. 400 x 300, <Bearbeiten> <als neue Auswahl einfügen>, nun steht das Bild zur Verfügung und kann gegebenenfalls bearbeitet werden (Ausschnitt erstellen, wenn das Bild nicht den ganzen Rahmen füllt, dann das eigentliche Bild ausschneiden und kopieren, und wieder wie oben in ein neues Bild einfügen. Dieses Bild nun auch lokal speichern.

4.3. Ein Bild aus dem Internet verwenden: die betreffende Seite mit dem Browser anzeigen, dann das Bild anklicken mit der rechten Maustaste, dann <Grafik speichern unter> auf dem lokalen PC.

5. Nun die Seite so aufbauen mittels einer Tabelle, dass das Bild in einem Feld der Tabelle eingefügt werden kann. Den Text entsprechend in die verschiedenen Felder einfügen. Das Ergebnis ist z.B. eine Tabelle mit 1 x 2 Feldern, das entsprechend mit Textteilen und Bildern gefüllt ist. Hierbei ist wichtig, dass man die Maße der Tabelle und Zellen in %-Werten des Fensters angibt (und nicht in Pixel). Damit ist gewährleistet, dass die Seite optimal auf dem Bildschirm des Anwenders erscheint. Der Anwender kann nämlich eine gute oder schlechte Grafikkarte besitzen oder auch einen größeren oder kleineren Bildschirm verwenden.

6. Weitere Verschönerungen am Text: z.B. Abschnitte einfügen, bestimmte Textteile fett machen oder die Schriftgröße ändern etc. Man kann auch bestimmten Textteilen eine andere Schriftfarbe geben.

7. Dieses Ergebnis lokal speichern und mit einem Browser (z.B. Firefox) anschauen.

8.1 Das fertige Programm mit ftp in ein spezielles Verzeichnis kopieren, die Bilder ebenso.

8.2 Alternativ: Nun diese fertige Seite mit dem Namen z.B. xxxx.html mit dem Windows-Programm write.exe (ich benütze lieber Winvi.exe) öffnen. Man sieht nun den Text einschließlich der nun vorhandenen html-Steuerzeichen. Diesen gesamten Text kopieren und in die Datenbanktabelle einfügen, z.B. im Administratormenü die Tabelle dokukataster öffnen <Einfügen> in das Datenbankfeld Text mittel <rechte Maustaste> <Einfügen> übernehmen.

9. In der Tabelle navigation kann dann der Aufruf eingefügt werden z.B. Katasterbeschreibung

Problem Bindestrich / Trennungsstrich

Es gibt in Word zwei verschiedene Striche, der eine ist die Taste rechts unten. Dieser Strich wird immer angezeigt, ist also in Wirklichkeit ein Bindestrich und kein Trennungsstrich. Der Trennungsstrich ist Control und -. Nun kommt es darauf an, wofür der Text entworfen ist.

1. Für Homepage html

Hier gibt es überhaupt keinen Trennungsstrich, das Minuszeichen rechts unten an der Tastatur ist immer ein Bindestrich oder ein Minuszeichen, also z.B. Augsburg-Kriegshaber, hier ist es ein Bindestrich, $3 - 2 = 1$, hier ist es ein Minuszeichen.

Einen Trennungsstrich gibt es hier überhaupt nicht, also keinen Trennungsstrich beim Schreiben mit Word verwenden, denn er wird als Bindestrich interpretiert. Das kommt daher, weil eine html-Datei keine feste Zeilenlänge hat, je nachdem, wie der Benutzer seinen Bildschirm eingestellt hat, hat derselbe Text einmal z.B. 10 Wörter pro Zeile, oder auch mehr oder weniger. Außerdem kann der Benutzer die Zeichengröße bestimmen, d.h. hier ändert sich wieder die Wortanzahl pro Zeile.

Ein - kann als Trennungsstrich dienen, z.B. wenn Augsburg-Kriegshaber so am Zeilenende zu stehen kommt, dass gerade noch Augsburg- hinpasst, dann kommt Kriegshaber auf die nächste Zeile.

Bei Formeln kann dies nicht gewünscht sein, also man will $3-2=1$ auf jeden Fall in einer Zeile haben. Hier gibt es auch Möglichkeiten, dies zu erzwingen. Man muss in diesem Falle einige Zeichen maskieren, z.B. die Zeichen Space (Zwischenraumzeichen), das Zeichen -. Bei dem normalen Space bzw. - würde der Browser evtl. einen Übertrag auf die nächste Zeile machen, was man aber verhindern will.

2. Für pdf-Datei (für Homepage oder auch zum Drucken)

In diesem Format bleibt die Zeilenbreite und auch die Zeichengröße, -type erhalten. Hier kann man Trennungsstriche verwenden, muss aber auch nicht.

3. Der Text ist ein Entwurf für eine Doktorarbeit oder eine Zeitschriftenveröffentlichung. Hier ist es so, dass der Text in ein festes Schema (definierte Zeichengröße, Zeichentype (Arial etc.), Zeilenbreite, Zeilenhöhe etc. eingefügt wird, das Programm (z.B. Indesign) bestimmt das Aussehen, natürlich kann man anschließend dies noch bearbeiten. Die Trennung macht Indesign selbstständig, aber man kann es auch nachher verändern. Hier ist es sinnvoll, überhaupt keine Trennungsstriche zu verwenden, weil Indesign dies sowieso selbstständig macht.

Die Darstellung des Datums

Häufige Felder sind solche, die ein Datum und/oder Datum und Uhrzeit enthalten. Dies ist in der Datenbank standardmäßig so gelöst: z.B. 2014-03-29 10:05

Dies ist zwar inzwischen europaweit zum Standard erklärt worden, jedoch benutzen die verschiedenen Länder nach wie vor ihre alten Standards. So ist in Deutschland z.B. 29.03.2014 üblich, in den angelsächsischen Länder 03-29-2014 oder so ähnlich. Ein weiterer Quasistandard ist folgende Darstellung: 29-Mar-2014, hier werden also die ersten drei Buchstaben der englischen Monatsnamen verwendet, die fast immer den deutschen Monatsnamen entsprechen.

Zumindest habe ich die Erfahrung gemacht, verschiedene Leute haben diesbezüglich unterschiedliche Wünsche. Dies kann man durch geeignete Programmierung auch erfüllen. Dies nur zur Einleitung.

Eine Möglichkeit, die ich meist verwende, ist diese: 29-Mar-2014. Nachdem diese Art der Darstellung nicht die default-Darstellung eines Datumfeldes ist, muss man etwas programmieren:

Programm zur Darstellung des Datums

```
<!DOCTYPE html PUBLIC „-//W3C//DTD HTML 4.01 Transitional//EN“>
<html xmlns=“http://www.w3.org/1999/xhtml“ xml:lang=“de“>
<head>
  <style type=“text/css“>
    @import url(„FORMATE.CSS“) screen;
  </style>
  <meta content=“text/html; charset=ISO-8859-0“ http-equiv=“content-type“>
  <title>Kirchenchor Kriegshaber Dreifaltigkeit</title>
  <meta name=“author“ content=“Heinz Wember Augsburg, php7, 04-Feb-2019“>
</head>
<body>
<left>
<br>
<center>
<big><big><strong>Chortermine in der Zukunft</big></big></strong>
</center><br>
<br>
<table style=“width: 100%px; text-align: left; margin-left: left; margin-right: auto;“ border=“1“ cellpadding=“2“ cellspacing=“2“>
  <tbody>
<?php
  include „../db/zugriff-dbchorkriegshaber.inc.php“;

  $monate = array(,01‘=>‘Jan‘, ,02‘=>‘Feb‘, ,03‘=>‘Mär‘, ,04‘=>‘Apr‘, ,05‘=>‘Mai‘, ,06‘=>‘Jun‘,
,07‘=>‘Jul‘,
,08‘=>‘Aug‘, ,09‘=>‘Sep‘, ,10‘=>‘Okt‘, ,11‘=>‘Nov‘, ,12‘=>‘Dez‘);
  $wochentage = array(0=>‘So‘, 1=>‘Mo‘, 2=>‘Di‘, 3=>‘Mi‘, 4=>‘Do‘, 5=>‘Fr‘, 6=>‘Sa‘);
  $abfrage = „SELECT
    DATE_FORMAT(datum, ‚%Y-%m-%d‘) AS ‚datumddmmyyyy‘, date_format(datum, ‚%w‘) AS
,wochentagnr‘,
    DATE_FORMAT(zeit, ‚%H:%i‘) AS zeithhmm,
    kirche, programm
```

```

FROM termine
WHERE datum >= CURDATE()
ORDER BY datum“;
$ergebnis = mysqli_query($db, $abfrage);
?>

```

```

<table style=“width: 100%; text-align: left; margin-left: left; margin-right: auto;“ border=“1“ cellpadding=“2“ cellspacing=“2“>

```

```

<caption></caption><tbody>
<tr>
<td style=“width: 08%;“>Datum</td>
<td style=“width: 05%;“>Zeit</td>
<td style=“width: 10%;“>Kirche</td>
<td style=“width: 50%;“>Programm</td>
</tr>
<?php
while($row = $ergebnis->fetch_assoc())
{
echo „<tr> „;
$datummysql = $row[„datumddmmyyy“];
$arrayjjjjmmtt = explode(„-“, $datummysql);
$varmonat = $arrayjjjjmmtt[1];
$wochentagr = $row[„wochentagr“];
$wochentag = $wochentag[$wochentagr];
$datumformatiert = $wochentag.“, „.$arrayjjjjmmtt[2].“-“.$monate[$varmonat].“-“.$arrayjjjjmmtt[0];
echo „<td style=width: 08%;> $datumformatiert<br>“;
echo „<td style=width: 05%;> $row[zeithhmm]</td>“;
echo „<td style=width: 10%;> $row[kirche]</td>“;
echo „<td style=width: 50%;> $row[programm] </td></tr>“;
}
mysqli_close($db);
?>
</tbody>
</table>
<hr>
<center>
<a target=“_blank“ href=“index.html“>Übersicht Kirchenchor</a>
</center>

<small>Änderungsstand: 12-Jun-2019</small>
</body>
</html>

```

Chortermine in der Zukunft

Datum	Zeit	Kirche	Programm
Do, 09-Apr-2020	19:00	Dreifaltigkeit	Gründonnerstag
Fr, 10-Apr-2020	15:00	Dreifaltigkeit	Karfreitag
Mo, 13-Apr-2020	10:00	Dreifaltigkeit	Ostermontag Haydn: Orgelsolomesse B-dur für Soli, Chor und Orchester; Mozart: Laudate dominum für Solo, Chor und Orchester;
Do, 01-Jan-2099		Änderungsstand:	db-Tabelle termine: 16-Feb-2020

[Übersicht Kirchenchor](#)

Änderungsstand: 17-Jun-2019

```

<table style="width: 100%px; text-align: left; margin-left: left; margin-right: auto;" border="1" cellpadding="2" cellspacing="2">
  <tbody>
<?php
include "../db/zugriff-dbchorkriegshaber.inc.php";
$abfrage = "SELECT
  CONCAT(
    CASE DATE_FORMAT(datum,'%w')
      WHEN 0 THEN 'So'
      WHEN 1 THEN 'Mo'
      WHEN 2 THEN 'Di'
      WHEN 3 THEN 'Mi'
      WHEN 4 THEN 'Do'
      WHEN 5 THEN 'Fr'
      WHEN 6 THEN 'Sa'
    ELSE 'was anderes' END,
    ',' ,
    DATE_FORMAT(datum,'%d.%m.%Y')
  ) as datumform, DATE_FORMAT(zeit, '%H:%i') as zeithhmm, kirche, programm, datum
FROM termine
ORDER BY datum DESC";
$ergebnis = mysqli_query($db, $abfrage);
?>

```

```

<table style="width: 100%; text-align: left; margin-left: left; margin-right: auto;" border="1" cellpadding="2" cellspacing="2">
  <caption></caption><tbody>
  <tr>
    <td style="width: 08%;">Datum</td>
    <td style="width: 05%;">Zeit</td>
    <td style="width: 10%;">Kirche</td>
    <td style="width: 50%;">Programm</td>
  </tr>
<?php
while($row = $ergebnis->fetch_assoc())
{
  echo "<tr> ";
  echo "<td style=width: 08%;> $row[datumform]<br>";
  echo "<td style=width: 05%;> $row[zeithhmm]</td>";
  echo "<td style=width: 10%;> $row[kirche]</td>";
  echo "<td style=width: 50%;> $row[programm] </td></tr>";
}
mysqli_close($db);
?>

```

Zugegebenermaßen ist diese Programmierung ein bisschen anstrengend, aber nur einmal, man kann es immer wieder bei anderen Programmen unverändert einfügen.
Das Ergebnis dieses Programms ist folgendes: Ausgabe hierzu

alle Chortermine in der Zukunft und Vergangenheit

Datum	Zeit	Kirche	Programm
Do, 01.01.2099		Änderungsstand:	db-Tabelle termine: 16-Feb-2020
Mo, 13.04.2020	10:00	Dreifaltigkeit	Ostermontag Haydn: Orgelsolomesse B-dur für Soli, Chor und Orchester; Mozart: Laudate dominum für Solo, Chor und Orchester;
Fr, 10.04.2020	15:00	Dreifaltigkeit	Karfreitag
Do, 09.04.2020	19:00	Dreifaltigkeit	Gründonnerstag
Mo, 06.01.2020	09:30	Thaddäus	Dreikönig: Mozart Missa brevis in B KV 275 für Sol, Chor und Orchester Gjeilo: Northern Lights (Pulchra es amica mea)
Do, 26.12.2019	10:00	Dreifaltigkeit	Weihnachten Karl Kempfer Pastoralmesse in G für Soli, Chor und Orchester; Karl Kempfer: Hodie_Christus_natus_est Gjeilo: Northern Lights (Pulchra es amica mea)
So, 20.10.2019	10:00	Dreifaltigkeit	Kirchweih: Mozart Missa brevis in B KV 275 für Sol, Chor und Orchester
So, 13.10.2019	17:00	Dreifaltigkeit	Kirchenkonzert 2019 Programm des gemischten Kirchenchores: Liedblatt 875, Ave Maria von Lonquich, Ave Maria von Rachmaninov, Northern Lights von Gjeilo, Ubi caritas von Snyder, Regina Coeli von Lotti, Ave verum corpus von Elgar, Blei bei uns Herr von Monk/L'Estrange, Ave Maria von Arcadelt; Frauen des Kirchenchores: Laudate pueri von Mendelssohn; Weitere Programmteile von den Chören Gaudeamus und Kinderchor
So, 16.06.2019	10:00	Dreifaltigkeit	Patrozinium: Mozart Missa brevis für Soli Chor und Orchester "Spatzenmesse" KV 220 C-dur; Knut Nystedt: Laudate dominum Mozart: Tantum ergo für Solo, Chor und Orchester KV 142 B-dur;
Mo, 22.04.2019	10:00	Dreifaltigkeit	Ostermontag Mozart: Krönungsmesse C-dur, KV 317 für Soli, Chor und Orchester; Mozart: Soli für S und T: Sub tuum praesidium F-dur KV 198; Händel: Halleluja aus dem Messias für Soli, Chor und Orchester;
Fr, 19.04.2019	15:00	Dreifaltigkeit	Karfreitag Chor und Volksgesang: Herr, auf dich vertraue ich (GL665); Chor und Volksgesang: Lass mich deine Leiden singen (GL733,1) Chor: J. S. Bach: O Haupt voll Blut und Wunden; Andreas Wilscher: Domine exaudi orationem meam Rihards Dubra: O crux ave Spiritual: Herr, ich möcht dein Leiden fassen; Chor und Volksgesang: Selig, wem Christum auf dem Weg begegnet (GL 275);

			Francesco Durante: Per signum crucis; D. Lorenzo Persosi: O salutaris hostia; Chor und Volksgesang: Meine Hoffnung meine Freude, meine Stärke, mein Licht (GL 365);
Do, 18.04.2019	19:00	Dreifaltigkeit	Gründonnerstag Chor und Volksgesang: Kyrie eleison (Vaticanum VIII) GL108; Chor: Wirf dein Anliegen auf den Herrn (Mendelssohn); Chor und Volksgesang: Wo die Güte und die Liebe wohnt (Salzburger Kirchenbuch); Chor und Volksgesang: Heilig, heilig, heilig (GL129); Chor und Volksgesang: Agnus Dei (Vaticanum XVII) GL 119 Chor: Hier ist das Haus des Herrn (Wolfram Menschik);
So, 06.01.2019	09:30	Thaddäus	Dreikönig K. Kempter: Orchestermesse Pastoralmesse in G; K. Kempter: Hodie Christus natus est; Gjeilo: Northern Lights (Pulchra es amica mea)
Mi, 26.12.2018	10:00	Dreifaltigkeit	Weihnachten Haydn: Orgelsolomesse B-dur für Soli, Chor und Orchester; Gjeilo: Northern Lights (Pulchra es amica mea)
So, 21.10.2018	10:00	Dreifaltigkeit	Kirchweih Mozart Orchester-Messe C-dur KV 259 Bach: Ehre sei Gott dem Herren, Alleluja Bach: O Jesu Christ, meins Lebens Licht
So, 27.05.2018	10:00	Dreifaltigkeit	Patrozinium Joseph Haydn: Missa Sancti Nicolai Hob. XXII:6 (deutsch Nikolaimesse); John Rutter: Gott sei stets in mir; Solo: Laudamus te aus Messe c-moll KV 427
Mo, 02.04.2018	10:00	Dreifaltigkeit	Ostermontag Mozart: Orchestermesse Krönungsmesse C-dur, KV 317; Solo: Laudamus te aus Messe c-moll KV 427
Fr, 30.03.2018	15:00	Dreifaltigkeit	Karfreitag Bach: O Haupt voll Blut und Wunden; Silcher: Schau hin nach Gogatha; Victoria: Populus meus; Dubra: O crux ave; Gjeilo Ola: Ubi caritas;
Do, 29.03.2018	19:00	Dreifaltigkeit	Gründonnerstag: Dubra: Kyrie eleison, Agnus dei, Mendelssohn: Auf Gott allein will hoffen ich; Salzburger-Kirchenbuch: Wo die Güte und die Liebe wohnt; Casali: Sanctus; Durufle: Vater unser;
Mi, 28.02.2018	17:00	Nikolaus Stadtbergen	Konzert: Mendelssohn: Jauchzet dem Herrn alle Welt; Mendelssohn-Waßner: Hebe deine Augen auf; Menschick: Gott hat seinen Engeln befohlen; Mendelssohn: Auf Gott allein will hoffen ich; Dubra: Ave Maria (II); Dubra: Panis angelicus; Gjeilo: Uni caritas; Durufle: Vater unser; Hirsh-Grill: Osseh shalom; Mendelssohn: Denn er hat seinen Engeln; Nystedt: Laudate;
Sa, 06.01.2018	09:30	Thaddäus	Dreikönig Mozart: Missa brevis in G, KV140; Alwin M. Schronen: Puer natus in Bethlehem
Di, 26.12.2017	10:00	Dreifaltigkeit	Weihnachten K. Kempter Pastoralmesse; K. Kempter Hodie Christus natus est; Alwin M. Schronen: Puer natus in Bethlehem

So, 05.11.2017	17:00	Dreifaltigkeit	Konzert Mendelssohn: Jauchzet dem Herrn alle Welt; Mendelssohn-Waßner: Hebe deine Augen auf; Mendelssohn: Auf Gott allein will hoffen ich; Menschick: Gott hat seinen Engeln befohlen; Dubra: Ave Maria (II); Laszlo: Jubilate Deo universa terra; Gjeilo: Ubi caritas; Dubra: Panis angelicus; Mendelssohn: Denn er hat seinen Engeln befohlen; Durufle: Vater unser; Nystett: Laudate;
So, 15.10.2017	10:00	Dreifaltigkeit	Kirchweih Mozart Orchester-Messe C-dur KV 259, Mendelssohn: Denn er hat seinen Engeln
So, 11.06.2017	10:00	Dreifaltigkeit	Patrozinium Haydn Nicolai-Messe; Mendelssohn: Jauchzet dem Herrn, alle Welt
Mo, 17.04.2017	10:00	Dreifaltigkeit	Ostermontag Mozart Spatzenmesse KV 220 C-dur; Mozart: Tantum ergo KV 142 B-dur;
Fr, 14.04.2017	15:00	Dreifaltigkeit	Karfreitag Salzburger-Kirchenbuch: Wo die Güte und die Liebe wohnt; Bach: O Haupt voll Blut und Wunden; Silcher: Schau hin nach Gogatha; Victoria: Populus meus; Dubra: O crux ave; Gjeilo Ola: Ubi caritas;
Do, 13.04.2017	19:00	Dreifaltigkeit	Gründonnerstag: Dubra: Kyrie eleison, Agnus dei, Panis angelicus; Mendelssohn: Auf Gott allein will hoffen ich; Salzburger-Kirchenbuch: Wo die Güte und die Liebe wohnt; Casali: Sanctus;
So, 19.02.2017	17:00	Thaddäus	Konzert: Mendelssohn: Hebe deine Augen auf; Mendelssohn: Auf Gott allein will hoffen ich; Menschick: Gott hat seinen Engeln befohlen; Laszlo: Jubilate Deo: Gjeilo: Ubi caritas et amor; Dubra: Ave Maria, gratia plena; Dubra: Panis angelicus; Nystedt: Laudate Dominum;
Fr, 06.01.2017	09:30	Thaddäus	DreiKönig K. Kempter: Pastoralmesse in G; Alwin M. Schronen: Puer natus in Bethlehem
Mo, 26.12.2016	09:00	Dreifaltigkeit	Weihnachten Mozart: Missa brevis in G, KV140; Alwin M. Schronen: Puer natus in Bethlehem
So, 30.10.2016		Thaddäus	Patrozinium Mozart Missa brevis in F KV 192; Laudate Dominum KV 339; Nystet: Laudate, laudate Dominum
So, 16.10.2016		Dreifaltigkeit	Kirchweih: Mozart Missa brevis in F KV 192; Laudate Dominum KV 339; Nystet: Laudate, laudate Dominum
So, 05.06.2016	10:00	Dreifaltigkeit	Pfarrfest Kirchenchor: Dubra: Kyrie, Agnus Dei; Nystedt: Laudate, laudate Dominum; Mendelssohn-Waßner: Hebe deine Augen auf; Menschick: Gott hat seinen Engeln befohlen;

Datenbank Design

Dies ist ein schwieriges Thema. Natürlich könnte ich auch etwas zur Theorie, Design der Datenbanken und deren Normalformen sagen, ich vermute aber, das wollen die meisten gar nicht wissen. Die meisten wollen sicher wissen, wie sie ihr Geschichtsthema in einer Datenbank abbilden können.

Ich habe es auch mehr mit Aristoteles als mit Platon gehalten, Aristoteles ist vom Beispiel zur Theorie vorgestoßen. Ich muss auch zugeben, dass ich fast alle neuen Programmiersprachen, Betriebssysteme und Datenbanken durch ein funktionierendes Beispiel gelernt habe. Also fangen wir mit einem sehr umfangreichen Beispiel an, die Ortsgeschichte eines Dorfes mit Kataster und Personen. Hier nun alle Tabellen dieser Datenbank. Es sind sehr viele gleichartige Tabellen dabei, auf die man nicht eingehen muss. Aber alle Tabellen, die voneinander abhängen, werde ich im Detail erläutern, warum sie so und nicht anders aufgebaut sind. Datenbank Kriegshaber: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/db-definition.txt>

1. Zunächst die unterschiedliche Tabellen-Typen

Alle Tabellen, die mit doku beginnen und noch viele andere auch, sind ganz einfache Tabellen, eigentlich sind es nur Textdateien. Um die muss ich mich gar nicht kümmern, das ist sicher jedem klar.

2. Die Tabelle haeuser

Dies ist in meinem Fall die Haupttabelle, um die sich alles dreht. Hier sind sehr viele Felder vorhanden, die alle eines gemeinsam haben, sie sind alle bezogen auf den Hauptschlüssel plannr. Wie ersichtlich, ist die plannr (Flur-Nummer) ein "unique key", also in der Tabelle kommt eine spezielle plannr nur einmal vor. Das ist auch in der Wirklichkeit so, in einer Steuergemeinde, so wie sie um 1808 in Bayern definiert wurde, hatte jedes Grundstück eine Nummer, die gleiche Nummer darf in dieser Steuergemeinde nicht noch einmal vorkommen. Wenn ein Grundstück geteilt wird, dann bekommen die abgeteilten Flächen wieder eine einzige neue Nummer. Z.B. Das Grundstück 437 (das ist der Exerzierplatz, der einmal 100 Tagwerk groß war), wird so nach und nach bebaut. Die neuen Grundstücke heißen 437 1/2, 437 1/3 bzw. 437/002 437/003, etc. Dadurch, dass man in dieser Tabelle das Feld plannr als unique definiert hat, wird die Eingabe einer weiteren bereits vorhandenen plannr verhindert. (Hier muss ich gleich ergänzen, 437/2, 437/02 oder 437/002 ist in Wirklichkeit immer dasselbe Grundstück, nachdem das Feld plannr vom Typ Character sein muss - wegen des / -, habe ich für meine Tabelle die Regel aufgestellt, dass die Nummer nach dem Strich pro Flur-Nummer immer gleich lang sein muss, damit bei einem Sort die Nummern schön nacheinander kommen.)

Es ist aber noch ein weiterer Schlüssel definiert, das ist die Hausnummer (Feld hausnr1913). Das habe ich deshalb so genannt, weil es eine Quelle Haus- und Straßenverzeichnis von 1913 gibt, von der ich ursprünglich ausgegangen bin. Wie jeder weiß, gibt es die Litera-Nummern in der Innenstadt von Augsburg seit 1938 nicht mehr, die Hausnummern der Vorstädte gibt es seit der Eingemeindung auch nicht mehr. Eine Auswertung war also vorgegeben, welche Hausnummer von 1913 hat heute welche Adresse - Straßename und Hausnummer. Dies sind die Felder strassennrheute und hausnrheute. (Nebenbei bemerkt, zur Systematik der Namen für die Feld- und Tabellennamen fällt auf, dass ich erstens nur Kleinbuchstaben verwende, deutsche Umlaute vermeide und auch Sonderzeichen kaum verwende. Das ist zwar nicht verboten, aber praktisch, man spart sich dadurch an anderer Stelle viel Arbeit. Mit anderen Worten, ich empfehle es, genauso zu verfahren.)

Das Feld strassennrheute ist ein wichtiges Feld, das natürlich noch in einer anderen Tabelle vorkommt. Es muss nicht genau so heißen, aber es muss genauso aufgebaut sein (numerisch). Eine andere Tabelle, die auch eine Straßenummer enthält, ist die Tabelle statamtstrassen (das ist eine Tabelle des Amtes für Statistik, die alle Straßen von Augsburg enthält). Ein Programm, das auf beide Tabellen, also haeuser und statamtstrassen zugreift, ist z.B. haeuserplannr.php. Der entscheidende Befehl aus diesem Programm ist folgender:

Verknüpfung Straßennummer mit Strassennamen

```
SELECT haeuser.plannr, haeuser.hausnr1913, haeuser.strassennr1913, haeuser.strassennr1918,
haeuser.strassennrheute, statamtstrassen.strname AS strassenname1918, haeuser.hausnr1918,
statamtstrassen.schluessel, haeuser.besitzer1913
FROM haeuser
LEFT JOIN statamtstrassen ON haeuser.strassennrheute = statamtstrassen.schluessel
ORDER BY plannr
```

Ich habe hier viele Felder weggelassen, die sind für die Theorie nicht wichtig. Was ich mit dem JOIN bezwecken will, habe ich weiter oben schon gesagt, ich wollte mir ersparen, den langen Strassennamen in jedem Satz für die Häuser einzugeben. Stattdessen habe ich zuerst die Tabelle statamtstrassen aufgenommen (die habe ich vom Amt für Statistik als excel-Tabelle bekommen), dann habe ich bei der Aufnahme der Häuser statt des Strassennamens die Straßennummer eingegeben. Also die Ulmer Straße hat den Schlüssel 1010, also habe ich jedes mal statt Ulmer Straße den Schlüssel 1010 erfasst. Der JOIN macht nun die Verknüpfung vom Schlüssel zum Straßennamen, das gewonnene Feld strassenname1918 wird gebildet durch die Verknüpfung der Tabellen haeuser mit statamtstrassen über das Feld strassennrheute (in der Tabelle haeuser) und schluessel (in der Tabelle statamtstrassen). Das Unangenehme bei diesem LEFT JOIN ist nun, wenn ich in der Tabelle haeuser eine Straßennummer eingegeben habe, die es in der Tabelle statamtstrassen nicht gibt, dann enthält das Ergebnis dieses SELECTs dieses spezielle Haus nicht, mit anderen Worten, fehlen alle vorkommenden Straßenschlüssel in der Tabelle statamtstrassen, bekomme ich überhaupt kein Haus angezeigt. Das darf natürlich nicht passieren. (Es gibt bessere Datenbanken, die dies automatisch verhindern, aber die Datenbank MySQL ist nicht so strukturiert). Wir brauchen also eine Methode, die verhindert, dass Häuser durch den Rost fallen. Alle 500 Häuser, die es 1918 bereits gab, einzeln abzuhaken, ist keine gute Lösung. Hier bringe ich gleich ein Programm, das mir solche Fehler anzeigt.

Dieses Prüfprogramm ist hauseval1918.php, das diesen wichtigen SELECT-Befehl enthält:

```
SELECT haeuser.ID, haeuser.plannr, haeuser.hausnr1913, haeuser.strassennr1918, haeuser.besitzer1913
FROM haeuser
LEFT JOIN statamtstrassen
ON haeuser.strassennr1918 = statamtstrassen.schluessel
WHERE statamtstrassen.schluessel IS NULL
ORDER BY plannr
```

Übrigens der Key hausnr1918 ist deshalb kein unique key, weil ein Haus mehrere Grundstücke haben kann. Nachdem es zu verschiedenen Zeiten unterschiedliche Hausnummern für ein Haus auf einer bestimmten plannr haben kann, gibt es außer dem Feld hausnr1918 noch die Felder hausnr1892, hausnr1867, hausnr1854, hausnr1840 und hausnr1810. Der Hintergrund dieser verschiedenen Hausnummern ist der, dass in Kriegshaber im 19. und 20. Jahrhundert sehr viel gebaut wurde, so dass unbebaute Flurnummern neue Häuser erhielten. Die Logik der Nummernvergabe war die, wenn ein Haus zwischen Hausnummer 80 und 81 gebaut wurde, bekam es die Nummer 80 1/2. Dies wurde mit der Zeit ziemlich unübersichtlich, so dass bei einer Revision des Grundsteuerkatasters die Nummern neu vergeben wurden, die dann keine Bruchnummern mehr hatten.

Quellendarstellung im Staatsarchiv.

Nun ein anderes Beispiel von Tabellen, die alle aufeinander bezogen sind:

Dies sind die Tabellen: katabkn, katband, katkopf, kattypr und katverz. Diese fünf Tabellen enthalten meine Quellen im Staatsarchiv. Der Grund ist der gleiche wie bei der Tabelle haeuser, die Werte in den Tabellen sollten redundanzfrei sein, mit anderen Worten alle Daten, die einmalig vorkommen, sollten auch nur einmal erfasst werden müssen. Der Archivar des Archivs hat sich dadurch beholfen, dass er gleichartige Daten auf ein Blatt geschrieben hat und dieses dann so oft wie nötig vervielfältigt hat. Die variablen Daten kamen dann von Hand dazu. Hier ein Beispiel eines Verzeichnisblattes für die Kataster von Kriegshaber.

Amt: Augsburg - Stadt
 Steuergemeinde: Kriegshaber
 Steuerdistrikt:
 vordem Augsburg - Land

Landgericht/Amtsgericht:
 vordem: Landgericht Augsburg

1 Besitzfassungen A-Land angelegt: <i>bf</i> <i>1809</i>	Nr. 32 I Fass. 1-113	32 II 184-334	
2 Häuser- und Rustikalsteuer- A-Land kataster, angelegt: <i>1710</i>	Nr. 12 <i>AL</i> Bes.Nr. 1-649		
3 Umschreibbuch zu 2 von bis	Nr. 12 <i>441</i>		
4 Grundsteuerkataster <i>gr 1</i> Urkataster abgeschlossen: <i>u/kg</i> <i>1840</i>	Nr. 48 Ks. 1-335 Hs.Nr. 1-Lude	N Ks. Hs.Nr.	N Ks. Hs.Nr.
5 Flächenrepertorium zu 4	Nr. <i>101</i>		
6 Umschreibgrundsteuer- kataster, gebunden 1832 von 1832 bis 1842 <i>u/kg</i>	Nr. 74		
7 Umschreibhefte zum Um- schreibgrundsteuerkataster von 1842 bis 1853 <i>u/kg</i>	Nr. 91 I Ks. 3-200 Hs.Nr. 1-70	Nr. 91 II Ks. 202-336 Hs.Nr. 71-Lude	Nr. Ks. Hs.Nr.
8 Grundsteuerkataster <i>gr 1</i> erste Renovation vom Jahre <i>1854</i>	Nr. 108 I Ks. 1-334 Hs.Nr. 1-Lude	Nr. Ks. Hs.Nr.	Nr. Ks. Hs.Nr.
Grundsteuerkataster <i>gr 2</i> zweite Renovation vom Jahre <i>1867</i>	Nr. 108 II Ks. 1- 334 508 Hs.Nr. 1-Lude	N Ks. Hs.Nr.	N Ks. Hs.Nr.
Grundsteuerkataster <i>gr 3</i> dritte Renovation vom Jahre <i>1892</i>	Nr. Ks. 1-1120 Hs.Nr. 1-Lude	N Ks. Hs.Nr.	N Ks. Hs.Nr.
9 Flächenrepertorium zur 1. Renovation, Abschluß <i>1854</i>	Nr. 177 I		
Flächenrepertorium zur 2. Renovation, Abschluß <i>1867</i>	Nr. 177 II		
Flächenrepertorium zur 3. Renovation, Abschluß <i>1892</i>	Nr. <i>1892-1940</i>	<i>1940-1961</i>	
10 Umschreibhefte zum ersten renovierten Grundsteuer- kataster, von 1854 bis 1867	Nr. 141 I Ks. 1-170 Hs.Nr. 1-99	Nr. 141 II Ks. 172-334 Hs.Nr. 201-Lude	Nr. Ks. Hs.Nr.
Umschreibhefte zum zweiten renovierten Grundsteuer- kataster, von 1867 bis 1892	Nr. 141 III Ks. 2-170 Hs.Nr. 1-74	Nr. 141 IV Ks. 171-340 Hs.Nr. 75-155	Nr. 141 V Ks. 341-508 Hs.Nr. 74-Lude
Umschreibhefte zum dritten renovierten Grundsteuer- kataster, von 1892 bis 1961	Nr. Ks. 2-110 Hs.Nr. 1-86	Nr. Ks. 112-240 Hs.Nr. 87-158	Nr. <i>12te u. 13te Band</i> Ks. 241-350 Hs.Nr. 159-259
11 Anmeldeprotokolle von 1832 bis 1962	Nr. 63 I <i>1832-1855</i>	Nr. 63 II <i>1856-1862</i>	Nr.
12 Umschreibverzeichnisse von 1862 bis 1909	Nr. 156	Nr.	Nr.
13 Messungsverzeichnisse von 1909 bis 1904	Nr. <i>siehe Verzeichnisse</i>	Nr.	Nr.
14 Gewerbesteuerkataster	Nr. Jhrg.	Nr. Jhrg.	Nr. Jhrg.
15 Umschreibbuch zu 14 von bis	Nr.	Nr.	Nr.

Hier sind augenscheinlich die Kopfdaten für alle unten stehenden Quellen gleich. Als zweites sieht man sofort, dass es zwar nur eine Besitzfassung für 1809 gibt, es gibt aber zwei Bände davon.

Ebenso ist es für die dritte Renovation des Grundsteuerkatasters, der 1892 fertiggestellt wurde. In diesem Fall ist es nur ein Buch.

Noch umfangreicher ist es mit den Umschreibheften zum dritten renovierten Grundsteuerkaster, hier stehen auf diesem Blatt nur drei Bände, die anderen fünf Bände stehen auf einem Zusatzblatt. Wenn diese Daten redundanzfrei gespeichert werden sollen, muss man erst die Systematik erschließen.

Um mir das zu erleichtern, habe ich auf dieser Kopie (das ist nicht das Original des StAA) die länglichen Bezeichnungen wie "Umschreibhefte zum Umschreibgrundsteuerkataster von 1842 bis 1853" durch geeignete Abkürzungen ergänzt, in diesem Fall als uugkh. Jetzt ist aber offensichtlich, dass dies nicht nur in Kriegshaber so war, sondern auch in anderen Orten, wenn auch einige Daten variabel waren, das sind diejenigen, die mit Kugelschreiber eingetragen sind, in unserem Beispiel also 1842 bis 1853.

Jetzt haben wir eine mehrfach verknüpfte Logik, wir haben Kopfdaten, Typdaten, wir haben Verzeichnisdaten und Banddaten. Ich kürze dies nun ab und schreibe gleich das Ergebnis der Systematik in Form des SELECT-Befehls auf:

```
SELECT katbandn.id AS id, katkopf.raort AS raort, katkopf.rentamt AS ra,
       katkopf.steuergemeinde AS stg, katkopf.steuerdistrikt AS std,
       katkopf.landamtsger AS ger,
       katabk.name, katabk.abk AS abk,
       katbandn.typ AS typ, kattyprn.gruppe AS gruppe,
       katbandn.bandnr AS bnr, katbandn.ksv AS ksv, katbandn.ksb AS ksb,
       katbandn.hv AS hv, katbandn.hb AS hb, kattyprn.jahr AS jahr, kattyprn.jahr AS jahr
FROM katbandn
LEFT JOIN katkopf ON katkopf.raort = katbandn.raort
LEFT JOIN kattyprn ON katbandn.typ = kattyprn.typ
LEFT JOIN katabk ON kattyprn.abk = katabk.abk
ORDER BY katkopf.steuergemeinde, kattyprn.gruppe, kattyprn.typ, katbandn.ksv, katbandn.bandnr
```

Letztlich will ich alle Bände wissen, die ich untersucht habe (katbandn), die länglichen Bezeichnungen möchte ich aber nur einmal schreiben (abk), deshalb muss jeweils zu jeder Abkürzung in der Tabelle band ein Satz mit der entsprechenden Abkürzung der Tabelle kattyprn vorhanden sein. Außerdem darf es keine leeren Sätze geben (Abfrage auf NULL). Nun ist noch dieses Ergebnis mit den Kopfdaten zu verknüpfen. Alle untersuchten Bände des Katasters (nebenbei auch Grundbücher und Ablösungskassenbücher) sind diese hier: <https://heinz-wember.de/MySQL-Kriegshaber/katbandn.php>

Wollte man dieses Quellenverzeichnis auf Excel oder Word erfassen, wäre dies sehr viel Arbeit gewesen. Durch dieses Skript mit den dazugehörigen fünf Tabellen ist man in der Lage, alle Kataster, Grundbücher und Ablösungskassenbücher nicht nur im Staatsarchiv Augsburg, sondern bayernweit damit zu erfassen. In diesem Fall habe ich mir die Prüfprogramme gespart, nachdem es so wenige Daten (zumindest für mein Thema Kriegshaber) sind, fällt es sofort auf, wenn ich bei den Tabellen katkopf, kattyprn, oder katabk einen Eintrag übersehen hätte.

Eine weitere Gruppe aufeinander bezogener Tabellen dienen zur Bevölkerungsstatistik.

Die Auswertung der Kirchenbücher hinsichtlich verschiedener Kriterien, u.a. wie weit sind die Hochzeitspaare, die in Kriegshaber geheiratet haben, von ihrem Geburtsort entfernt.

Ausgehend vom Geburtsort der Hochzeiter soll eine Statistik erstellt werden, wie weit sie von ihrem Geburtsort entfernt sind.

In der Tabelle hochzeiten sind die Geburtsorte der Paare eingetragen.

In der Tabelle ortkreis ist die Zuordnung der Orte zu Kreisen enthalten.

In der Tabelle kreisentfernung wird pro Kreis die Entfernung der Kreisstadt zu Kriegshaber eingetragen.

```
CREATE TABLE `kreisentfernung` (  
  `kreis` varchar(255) COLLATE latin1_german1_ci NOT NULL,  
  `entfernung` int(7) DEFAULT NULL,  
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,  
  PRIMARY KEY (`kreis`),  
  KEY `entfernung` (`entfernung`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Kreis Entfernung';
```

```
CREATE TABLE `ortkreis` (  
  `ort` varchar(255) COLLATE latin1_german1_ci NOT NULL,  
  `kreis` varchar(255) COLLATE latin1_german1_ci DEFAULT NULL,  
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,  
  PRIMARY KEY (`ort`),  
  KEY `kreis` (`kreis`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Ort Kreis';
```

```
CREATE TABLE `hochzeiten` (  
  `mann` varchar(255) COLLATE latin1_german1_ci NOT NULL,  
  `jahr` int(4) DEFAULT NULL,  
  `frau` varchar(255) COLLATE latin1_german1_ci NOT NULL,  
  `mnachname` varchar(100) COLLATE latin1_german1_ci NOT NULL,  
  `mvorname` varchar(100) COLLATE latin1_german1_ci NOT NULL,  
  `mvornamegruppe` varchar(100) COLLATE latin1_german1_ci DEFAULT NULL,  
  `mberuf` varchar(100) COLLATE latin1_german1_ci DEFAULT NULL,  
  `mgebort` varchar(100) COLLATE latin1_german1_ci DEFAULT NULL,  
  `mgebjahr` int(4) DEFAULT NULL,  
  `mwitwer` varchar(1) COLLATE latin1_german1_ci NOT NULL DEFAULT 'n',  
  `fnachname` varchar(100) COLLATE latin1_german1_ci NOT NULL,  
  `fvorname` varchar(100) COLLATE latin1_german1_ci NOT NULL,  
  `fvornamegruppe` varchar(20) COLLATE latin1_german1_ci DEFAULT NULL,  
  `fberuf` varchar(100) COLLATE latin1_german1_ci DEFAULT NULL,  
  `fgebort` varchar(100) COLLATE latin1_german1_ci DEFAULT NULL,  
  `fgebjahr` int(4) DEFAULT NULL,  
  `fwitwe` varchar(1) COLLATE latin1_german1_ci NOT NULL DEFAULT 'n',  
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,  
  KEY `mgebort` (`mgebort`),
```

```

KEY `fgebort` (`fgebort`),
KEY `mvorname` (`mvorname`),
KEY `fvorname` (`fvorname`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci
COMMENT='Hochzeitsbuch Kriegshaber 1600-1975';

```

Folgender SELECT-Befehl ermittelt die Anzahl der Männer, die im Hochzeitsbuch von Kriegshaber (bzw. von Oberhausen mit Wohnort Kriegshaber) verzeichnet sind, die zwischen 1 und 20 km entfernt geboren wurden.

```

SELECT ortkreis.kreis AS kreis1, kreisentfernung.entfernung AS entfl, count(*) AS maennerimkreis,
SUM(kreisentfernung.entfernung) AS maennerimkreissum
FROM hochzeiten
LEFT JOIN ortkreis ON hochzeiten.mgebort = ortkreis.ort
LEFT JOIN kreisentfernung ON ortkreis.kreis = kreisentfernung.kreis
WHERE kreisentfernung.entfernung > 0
AND kreisentfernung.entfernung < 21
GROUP BY kreis1

```

Der erste JOIN verknüpft die Geburtsorte in der Tabelle hochzeiten mit der Tabelle ortkreis, die die Kreisstadt ermittelt.

Der zweite JOIN verknüpft die Tabelle ortkreis mit der Tabelle kreisentfernung, die die Entfernung der Kreisstadt von Kriegshaber enthält.

Der WHERE-Befehl wählt nur diejenigen aus, wo die Entfernung zwischen 1 und 20 km liegt.

Worauf kommt es bei der Definition der Felder in den einzelnen Tabellen an?

- Es ist wichtig, dass die Felder, bei denen der Geburtsort unbekannt ist, den Wert NULL hat.
- Ebenso ist es wichtig, dass bei einem unbekanntem Hochzeitsjahr der Wert NULL ist.
- Die Felder Witwer/Witwe (mwittwer, fwitwe) müssen immer ausgefüllt sein, entweder mit j, also Witwer/Witwe oder mit n, d.h. ledig vor der Hochzeit.
- Für weitere Felder, die ebenfalls ausgewertet werden, wie Vorname, Vornamengruppe und Beruf gilt ähnliches. (Dies wird hier aber nicht behandelt, das sind einfachere Auswertungen.)
- Das gleiche gilt für die Tabelle ortkreis, hier müssen die Felder ort und kreis ebenfalls ausgefüllt sein, notfalls mit dem Wert unbekannt.
- Schließlich muss in der Tabelle kreisentfernung der Wert für die Entfernung NULL sein, wenn man den Kreis nicht kennt. Ein unbekannter Kreis hat keine Entfernung, d.h. solche Sätze können bei der Auswertung nicht herangezogen werden, 100 % der Personen sind solche, wo ein Geburtsort, eine Kreisstadt und die Entfernung der Kreisstadt von Kriegshaber bekannt ist.

Das Ergebnis dieser Abfrage ist:

Summe der Entfernung vom Geburtsort der Männer - 001 bis 020 km

Kreis	Einzelentfernung	Anzahl Männer	Männer mal Entfernung
Augsburg-Kreis	20	236	4720
Augsburg-Stadt	5	68	340
Friedberg	15	18	270
Summe		Anzahl Männer	Männer mal Entfernung
Summe	Entfernung 1 bis 20 km	322	5330

[Übersicht Kirchenbücher](#)

Das komplette Programme für diese Auswertung mrimkrse001bis020.php:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url(../FORMATE.CSS) screen;
  </style>
  <meta name="author" content="Heinz Wember Augsburg, php7, 25-Feb-2019">
  <title>Summe der Entfernung vom Geburtsort der Männer - 001 bis 020 km</title>
</head>
<body>
<br><center>
<h1>Summe der Entfernung vom Geburtsort der Männer - 001 bis 020 km</h1>
</center><left>
<br>
<?php
  define(„UNLOCK“,1);
  include „../db/zugriff-dbkriegshaber.inc.php“;
  $abfrage1 = „SELECT ortkreis.kreis AS kreis1, kreisentfernung.entfernung AS entf1, count(*) AS maenne-
rimkreis, SUM(kreisentfernung.entfernung) AS maennerimkreissum
    FROM hochzeiten
    LEFT JOIN ortkreis ON hochzeiten.mgebort = ortkreis.ort
    LEFT JOIN kreisentfernung ON ortkreis.kreis = kreisentfernung.kreis
    WHERE kreisentfernung.entfernung > 0
    AND kreisentfernung.entfernung < 21
    GROUP BY kreis1“;
  $ergebnis1 = mysqli_query($db, $abfrage1);
  $abfrage2 = „SELECT count(*) AS maennerimkreis2, SUM(kreisentfernung.entfernung) AS maennerim-
kreissum2
    FROM hochzeiten
    LEFT JOIN ortkreis ON hochzeiten.mgebort = ortkreis.ort
    LEFT JOIN kreisentfernung ON ortkreis.kreis = kreisentfernung.kreis
    WHERE kreisentfernung.entfernung > 0
    AND kreisentfernung.entfernung < 21“;
  $ergebnis2 = mysqli_query($db, $abfrage2);
?>

<table style="width: 50%; text-align: left; margin-left: left; margin-right: auto;" border="1" cellpadding="2"
cellspacing="2">
  <caption></caption><tbody>
    <tr>
      <td style="width: 20%;">Kreis</td>
      <td style="width: 10%;">Einzelentfernung</td>
      <td style="width: 10%;">Anzahl Männer</td>
      <td style="width: 10%;">Männer mal Entfernung</td>
    </tr>

<?php
  while($row = $ergebnis1->fetch_assoc())
  {
    echo „<tr><td style=width; 20%;>$row[kreis1]</td>“;
```

```

    echo „<td style=width; 10%;>$row[entfl]</td>“;
    echo „<td style=width; 10%;>$row[maennerimkreis]</td>“;
    echo „<td style=width; 10%;>$row[maennerimkreisum]</td></tr>“;
}
?>

```

```

<table style=“width: 50%; text-align: left; margin-left: left; margin-right: auto;“ border=“1“ cellpadding=“2“
cellspacing=“2“>

```

```

<caption></caption><tbody>
<tr>
<td style=“width: 20%;“>Summe</td>
<td style=“width: 10%;“> </td>
<td style=“width: 10%;“>Anzahl Männer</td>
<td style=“width: 10%;“>Männer mal Entfernung</td>
</tr>

```

```

<?php
while($row = $ergebnis2->fetch_assoc())
{
    echo „<td style=width; 20%;>Summe </td>“;
    echo „<td style=width; 10%;>Entfernung 1 bis 20 km </td>“;
    echo „<td style=width; 10%;>$row[maennerimkreis2]</td>“;
    echo „<td style=width; 10%;>$row[maennerimkreisum2]</td>“;
}
mysqli_close($db);
?>

```

```

</tbody>
</table>

```

```

<hr>
<center>
<a target=“_blank“ href=“../index-kb.html“>Übersicht Kirchenbücher</a>
</center>
<hr>
<small>Änderungsstand: 14-Mai-2019<br>
<a href=“../Impressum.html“>Heinz Wember</a>
</small>
</body>
</html>

```


Neues Grundbuch Ablösungskasse

Hier sind die Tabellen ngbdetart (und ngbdetail) mit zwei weiteren Tabellen verknüpft: die Tabelle ngbschl, das den Lehensträger anzeigt und die Tabelle ngbkr gem, die die Umschlüsselung des Kreisgemeindeschlüssels zu Kreis und Gemeinde vornimmt.

```
SELECT kreis, gemeinde, katseite, plannr, hausnr, ngbdetart.schl, art,  
fgfl, fgkr, fghl, bemerkung, ngbschl.besitzer AS besitzer  
FROM ngbdetart  
LEFT JOIN ngbschl ON ngbschl.schl = ngbdetart.schl  
LEFT JOIN ngbkr gem ON ngbkr gem.schl = ngbdetart.stgem  
ORDER BY stgem, hausnr
```

Password geschützte Verzeichnisse auf der Homepage

Ich möchte einige Kommentare abgeben, wie und warum es sinnvoll ist, Password geschützte Verzeichnisse auf der Homepage zu haben.

1. Schutz der Daten, die nach dem Gesetz nicht zu veröffentlichen sind.

Das Gesetz zum Schutz von persönlichen Daten ist erst vor kurzer Zeit neu definiert worden. Es sind persönliche Daten von Personen zu schützen, die noch nicht 30 Jahre tot sind. Wenn es unbekannt ist, ob eine Person noch lebt, dann gilt die Regel, die Daten von Personen, die vor weniger als 110 Jahre geboren sind, sind zu schützen. Eine dritte Regel ist, die persönlichen Daten von Personen, die vor weniger als 80 Jahren geheiratet haben, sind zu schützen.

Ausgehend von diesem Gesetz habe ich die Daten des Katasters, speziell die Umschreibhefte zum Kataster 1892, für diesen Personenkreis unter Passwortschutz gestellt. Natürlich gilt für die meisten Eintragungen in diesen Umschreibheften dieser Schutz nicht, jemand, der um 1900 ein Haus gekauft oder geerbt hat, ist in der Regel schon mehr als 30 Jahre tot. Nachdem aber diese Umschreibhefte zum Teil Nachträge bis 1962 enthalten, ist es möglich, dass diese Personen noch leben bzw. noch keine 30 Jahre tot sind. Nachdem ich in Kriegshaber lebe, kenne ich sehr viele Leute von Kriegshaber, zumindest ist mir eine Person bekannt, die 1954 das Haus von der Mutter geerbt hat und noch lebt. In diesem Fall wäre es zwar kein Problem gewesen, nachdem diese Person der Veröffentlichung dieser Katasterdaten zugestimmt hätte, aber es sind noch viele andere Einträge vorhanden, für die der Schutz zutreffen würde. Nachdem es zu aufwändig ist, alle diese lebenden Personen zu fragen, bzw. von diesen Personen das Todesdatum zu ermitteln, habe ich alle diese Umschreibhefte unter Passwortschutz gestellt.

2. Schutz von eigenen Daten

Daneben sind zur Zeit sehr viele Auswertungen von Quellen des Staatsarchivs, des Stadtarchivs und des Bistumsarchivs von Augsburg Password geschützt auf meiner Homepage. Dies hat den einfachen Grund, dass ich diese Informationen in nächster Zeit in einem Buch veröffentlichen möchte. Es ist klar, dass ich kaum gewinnbringend ein Buch veröffentlichen kann, dessen Daten seit Jahren schon im Internet veröffentlicht sind.

3. Daten, die es gestatten, Informationen auf der Homepage abzufragen, die zu schützen wären.

Hier handelt es sich im Wesentlichen um die Einlogginformationen zu meinen Datenbanken. Neben der Datenbank kriegshaber habe ich noch weitere 10 MySQL-Datenbanken, die im Internet auf meinem Server gespeichert sind, wo alle Informationen Password geschützt stehen.

Speziell diese Dateien stehen auf einem Password geschützten Verzeichnis. Nachdem dies wohl für die Leser interessant ist, wie man das machen kann, wenn sie ebenfalls daran denken, ihre Forschungen in einer Datenbank im Internet zu speichern, zeige ich kurz das Prinzip auf.

Bei allen Programmen, die Daten aus einer Datenbank lesen, ist eine Eröffnungsprozedur für diese spezielle Datenbank notwendig. Diese Prozedur habe ich bei allen Programmen ausgelagert, so dass ich meine Skripten, die auch hier veröffentlicht sind, ohne weiteres offen zeigen kann. Der wesentliche Befehl ist hier:

Der Befehl, auf den es ankommt, ist der include. Die Informationen zum Öffnen der Datenbank stehen im Verzeichnis db, das mit einem Passwort, das nur ich kenne, geschützt ist. Jedes Programm, das auf diese Datei mit dem Namen zugriff-dbkriegshaber.inc.php von außen zugreift, wird der Zugang verwehrt. Alle Programme, die auf meiner Homepage stehen, können aber ohne weiteres darauf zugreifen.

Hier das Muster dieser Datei:

```
<?php
define(„UNLOCK“,1);
include „../db/zugriff-dbkriegshaber.inc.php“;
$abfrage = „SELECT haeuser.hausnr1913, haeuser.plannr, haeuser.strassenr1913, haeuser.strassenr1918,
haeuser.strassennrheute, statamtstrassen.strname AS strassenname2012,
```

```

statamtstrassen.schluessel ,haeuser.besitzer1913, haeuser.hausnrheute,
haeuser.hausnr1892, haeuser.besitzer1892, haeuser.hausnr1867, haeuser.besitzer1867,
haeuser.hausnr1840, haeuser.besitzer1840, haeuser.hausnr1810, haeuser.besitzer1810
FROM haeuser
LEFT JOIN statamtstrassen ON haeuser.strassenrheute = statamtstrassen.schluessel
ORDER BY strassenname2012, haeuser.hausnrheute, plannr";

```

```

$ergebnis = mysqli_query($db, $abfrage);
?>

```

```

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head> <meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
<title>Connect to MySQL-DB</title>
</head>
<body>
<?php

```

```

define ( ,MYSQL_HOST' , ,localhost' );
define ( ,MYSQL_BENUTZER' , ,dbo' );
define ( ,MYSQL_KENNWORT' , ,password' );
define ( ,MYSQL_DATENBANK' , ,db' );

```

```

$db = mysqli_connect(MYSQL_HOST, MYSQL_BENUTZER, MYSQL_KENNWORT, MYSQL_DA-
TENBANK);

```

```

if ($db )
{
}
else
{
echo „Fehler: konnte nicht mit MySQL verbinden.“ . PHP_EOL;
echo „Debug-Fehlernummer: „ . mysqli_connect_errno() . PHP_EOL;
echo „Debug-Fehlermeldung: „ . mysqli_connect_error() . PHP_EOL;
exit;
}
?>
</body>
</html>

```

Diese vier Informationen zu der Datenbank HOST, BENUTZER, KENNWORT und DATENBANK sind der wesentliche Inhalt. Jeder, der eine Datenbank lokal auf seinem Server erstellt, bekommt vom System diese Daten mitgeteilt. In meinem Fall stehen diese Datenbanken auf meinem Server, auf den nur ich zugreifen kann, d.h. alle Daten auf diesem PC, der physisch im Rechenzentrum der Firma lundl steht, gehören mir, kein anderer hat Zugriff, wenn ich es nicht ausdrücklich erlaube.

Nicht jeder braucht für seine Anwendung so viel Platz wie ich, z.B. hier ist es dann so, dass auf einem physischen PC Daten stehen, der von mehreren Personen benützt wird. Auch hier will ich Beispiel bringen:

```

define ( ,MYSQL_HOST' , ,localhost' );
define ( ,MYSQL_BENUTZER' , ,dbo' );
define ( ,MYSQL_KENNWORT' , ,password' );
define ( ,MYSQL_DATENBANK' , ,db' );

```

Wie man sieht, ist dieses Skript praktisch identisch, nur der Parameter HOST ist anders, hier sind die Angaben zum physischen PC bei der Firma 1und1 angegeben.

Was die Sicherheit betrifft, sind diese beiden Möglichkeiten, die Datenbank entweder auf einem eigenen Server oder auf einem Server der Firma 1und1 zu speichern, identisch. Der Hauptunterschied besteht im Preis, der billigste Tarif bei 1und1 für einen persönlichen Server mit etwa 140 GB Plattenspeicher ist etwa 40 €, wenn man außer den 30 € für den Webspace noch die anderen Leistungen der Firma mitrechnet, wie Kosten für den Domainnamen (in meinem Fall also heinz-wember.de), oder die Virenprüfung für den E-Mail-Verkehr und schließlich die Kosten für die erweiterte Zugangsauswertung. Diese Zugangsauswertung erlaubt es dem Eigner eines Webspaces, jeden Zugriff auf die seine Domain auszuwerten.

Nebenbei bemerkt, ist der Preis für einen "gesharten" Server mindestens 7 € pro Monat, je nach weiteren Leistungen.

Wie allgemein bekannt, ist dies neben der aufgerufenen Seite, die IP-Nummer desjenigen, der zugegriffen hat mit Datum- und Zeitstempel. Was diese Speicherung der IP-Nummer für eine längere Zeit betrifft, wird zur Zeit hierüber noch gestritten, ob überhaupt und wenn ja, wie lange solche Daten beim Provider vorhanden sein müssen. Beim Gesetzentwurf der deutschen Regierung ist eine Speicherung von sechs Monaten Pflicht.

Die Kenntnis dieser IP-Nummer, die der Eigner des Webspaces kennt, ist im allgemeinen wertlos, weil diese Nummer in der Regel nicht fest ist, d.h. die Nummer wird vom Provider bei jedem Einloggen in das Internet neu vergeben. (Das hatte den einfachen Grund, dass es nicht so viele IP-Nummern alter Art - vier Byte - gibt, wie es Personen auf der Welt gibt.) Was man als Eigner einer Homepage natürlich weiß, ist, welche Bereiche der IP-Nummer wem gehören. Wer aber zur einer bestimmten Zeit welche IP-Nummer hat, weiß nur der Provider. Das Gesetz sieht vor, dass per Gerichtsbeschluss der Provider verpflichtet werden kann, den Namen und die Adresse des Benutzers seiner IP-Nummer mitzuteilen. So weit so gut. Wenn man von Abzockern eine Mahnung bekommt, man hätte unerlaubt etwas runtergeladen, ist dieses Schreiben wirkungslos, wenn der Provider nicht per Gerichtsbeschluss verpflichtet wurde, die detaillierten Daten des Nutzers mitzuteilen.

4. Untermenü mit Passwortschutz

Die einfachste Art ist die, diejenigen Programme, für die Passworteingabe Pflicht ist, in ein Verzeichnis zu kopieren, das über das Administratormenü mit Passwort zu versehen ist. Hier erfolgt automatisch vom Apache-Server eine Aufforderung, Benutzername/Passwort einzugeben.

Eine Alternative zum vorigen Schutzmechanismus ist folgende:

Während ich bei der vorigen Methode vorgesehen habe, dass es mehrere Benutzername/Passwort geben kann, habe ich speziell für den Administrator vorgesehen, dass dieser Schutz über ein anderes Verzeichnis läuft, für das nur der Administrator das Passwort hat.

Tipps zur Programmierung und zum Testen der Programme

Ich hoffe, dass diese Tipps für diejenigen von Nutzen sein werden, die angeregt durch mein Buch diese Methoden für die Veröffentlichung ihrer Forschungsergebnisse nutzen möchten.

1. Parallele Nutzung eines Laptops und der persönlichen Homepage

Weiter oben habe ich dies bereits angedeutet, ohne die Details zu sagen, welches kleine Programm man ändern muss, damit dies funktioniert. Nachdem ich im Kapitel Password gezeigt habe, wie man in eine Datenbank reinkommt, ist es auch jetzt schon klar, welches kleine Programm hier gemeint ist. Es sind vom Zugriffs-Skript nur die vier Parameter HOST, BENUTZER, KENNWORT und DATENBANK zu ändern, dann läuft die Applikation, ohne dass man sonst noch etwas zu ändern hätte. Ich habe mir angewöhnt, auf meinem PC drei Verzeichnisse zu haben, um dies zu steuern: db, db-lokal und db-internet. Natürlich kommen die Zugriffs-skripten für das Internet in das Verzeichnis db auf dem Server, die Skripten db-lokal kommen in das Verzeichnis db des Laptops, auf dem der Apache- und MySQL-Server liegt. Damit ist alles geregelt. Ich persönlich arbeite mit mehreren PCs, auf dem Desktop habe ich keinen MySQL-Server, der steht nur auf dem Laptop, den ich in die Archive mitnehme. Im Archiv schreibe ich meine gelesenen Quellen sofort in eine passende Tabelle der Datenbank. Zu Hause lese ich die erfassten Daten nochmals Korrektur, dann exportiere ich die geänderte DB und importiere sie in die Datenbank meiner Homepage. Nachdem ich seit Jahrzehnten mit Datenbanken arbeite, weiß ich immer ganz genau, welche Tabellen ich geändert habe, ich exportiere und importiere nur die geänderten Tabellen. Einem Anfänger würde ich aber raten, jeweils die gesamte Datenbank zu transferieren. Nicht zu vergessen sind aber die Sicherheitskopien. Meine Export-Datei heißt z.B. kri-20140329.sql, am Vortag hieß sie natürlich anders und diese Dateien hebe ich auf, nicht nur auf dem lokalen PC, sondern auch auf meinem freistehenden Magnetplatten, die normalerweise nicht mit dem PC verbunden sind. Während ich mit dem Internet verbunden bin, ist grundsätzlich keine Sicherheitsplatte am PC angeschlossen. Selbst wenn man jedes Update des Betriebssystems und der Antivirensoftware durchführt, ist ein Angriff auf seinen privaten PC möglich, es ist auch mir schon mehrmals passiert. Sehr unangenehm sind solche Angriffe, wenn man den Virus etc. nicht kennt und folglich auch nicht entfernen kann. Dies ist mir erst vor einigen Monaten passiert. Damit komme ich zum zweiten Tipp, wie organisiere ich meine persönlichen Daten auf dem PC. Im Falle eines solchen als schlimmsten anzunehmenden Fall, kann ich nur meine Festplatten des PCs formatieren und alles wieder neu installieren. Nachdem alle Programme neu installiert sind, kopiere ich meine drei gesicherten Verzeichnisse zurück, das dann wohl eine Nacht dauert. Merke: um einen PC installieren zu können, braucht man einen zweiten PC, der internetfähig ist, damit man die aktuellen Driver auch bekommt.

2. Die Datenorganisation auf dem eigenen PC.

Was sich inzwischen rum gesprochen haben müsste, ist eine konsequente Sicherung seiner persönlichen Daten. Man darf sich nicht täuschen, selbst wenn heutzutage die Festplatten extrem sicher sind, ich hatte in den letzten 10 Jahren nie einen "Crash", der in den 80er Jahren noch Gang und Gäbe war, aber es gibt auch andere Gründe, warum man seine Daten regelmäßig sichern sollte.

Als erstes würde ich sagen, man sollte die Daten nicht auf einer DVD sichern, die ist unzuverlässig. Hier hat man häufig Probleme, entweder mit der Silberscheibe oder mit dem DVD-Lesegerät, beides kann tödlich sein. Man hat ja nicht nur seine Forschungen auf dem PC, sondern auch seine digitalen Fotos, Filme, E-Mail, Briefe (die man in der Regel für sich gar nicht mehr ausdruckt) etc.

Wie mache ich dies?

Ich habe meine Daten, damit meine ich wirklich meine Daten und nicht das Betriebssystem und die installierten Programme, in separaten Verzeichnissen. Ich persönlich sichere nie das Betriebssystem oder meine installierten Programme. Ich weiß zwar, dass es entsprechende Backup-Systeme gibt, ich nutze sie aber nie, das ist mir zu zeitraubend. Es schadet dem PC auch nicht, wenn er einmal im Jahr neu installiert wird, er ist nachher viel munterer, zumindest gilt dies für Windows.

Wichtig ist, dass man seine eigenen Daten unter einem eigenen Verzeichnis hat, ich habe deren mehrere. Zum einen das Verzeichnis Daten, das alles enthält außer den genannten Ausnahmen, also die Exporte der Datenbank, die eigenen Programme der Homepage, die Briefe und auch meine E-Mail. Normalerweise, wenn

man einen E-Mail-Client hat, sind diese Daten unter Benutzer und den eigenen Daten gespeichert. Ich habe überhaupt keine persönlichen Daten in diesen Verzeichnissen unter Benutzer, ich kann mir das leisten, außer mir ist keiner auf dem PC. Man muss bei der Installation des E-Mail-Servers (ich habe Thunderbird) darauf achten, dass diese Informationen unter dem Verzeichnis Daten stehen. Dann habe ich noch weitere persönliche Verzeichnisse wie Fotos/Filme und Genealogiedatenbanken. Diese drei Verzeichnisse stehen unverändert, also ungepackt, auf den Sicherheitsplatten. Dazu benutze ich erstmalig den Copy des Betriebssystems, dann aber das Programm Allway Sync. Wer sich für dieses Programm interessiert, kann dies auf meiner Homepage nachlesen: <https://heinz-wember.de/dokus/html/Sicherheitskopien.html>. Für kleine Datenmengen ist das Programm kostenlos, bei größeren Datenmengen kostet es einmalig einen Betrag unter 100 €.

3. Testen mit php/MySQL

Ich schreibe hier nur auf, wie ich es mache, natürlich gibt es auch andere Meinungen, die durchaus auch richtig sind.

Die Programme schreibe ich mit dem Programm WinVi32 für Windows, das praktisch identisch mit dem vi unter Unix ist. Den vi kenne ich seit Jahrzehnten, er ist vielleicht nicht so einfach wie write.exe, aber man hat mehr Möglichkeiten. In der Regel ist das Schreiben der Programme ein Kopieren mit ein paar Änderungen. Das Grundgerüst, wie Sie bereits an den Beispielen gesehen habe, ist immer gleich. Was unterschiedlich ist erstens der SELECT. Diesen SELECT schreibe ich als erstes und teste ihn gleich mit der Datenbank unter phpmysql aus. In der Regel bin ich beim Programmieren im Internet eingeloggt und habe Zugriff auf die Administratorfunktionen meiner Datenbank bei lund1. Dazu braucht man natürlich ein Programm, das die Daten auf den Server bringt, ich benütze dazu sftp (nicht ftp aus gutem Grund!). Auf eines möchte ich dringend hinweisen, die Administratorfunktion bei lund1 ist auf eine Stunde beschränkt, dann wird ohne Warnung die Verbindung gekappt. Natürlich kann man sich danach wieder einloggen für eine Stunde. Also dran denken, nach etwa 50 Minuten ausloggen und gleich wieder einloggen. Am besten selbst alle Browser-Instanzen abbrechen, bevor man abgebrochen wird. Es ist mehr als ärgerlich, wenn alle seit einer Stunde eingegebenen Daten nicht gespeichert sind.

Nachdem der SELECT getestet und als ok befunden wurde, das Programm testen. Ich mache es so, dass das Programm mittels sftp auf den Server kommt, dann ergänze ich im Navigationsprogramm den Aufruf dieses Programms. In der Regel teste ich ohne Password, wenn also jemand so selben Zeit mein Programm aufruft, dann kann er die halb erfasste Quelle bereits lesen. Wenn man das nicht möchte, gibt es zwei sichere Verfahren:

- Das Programm auf ein Verzeichnis kopieren, bei dem man nur selbst das Passwort kennt oder
- Ein modifiziertes Navigationsprogramm verwenden. Wie ersichtlich frage ich bei jedem Eintrag in der Tabelle navigation ab, ob aktiv oder nicht. Ich kann also zum Testen nicht die Standard-index.html benützen, sondern das Programm test.html, das einen Navigationsprogrammaufruf benützt, der alle Einträge ausführt, also auch die inaktiven.

Wenn beim Ausführen eines php-Programms ein Fehler auftritt, wird die Zeile, in dem der Fehler war, mit einem Kommentar angezeigt. Daher auch mein Rat, vi bzw. WinVi32 zu benutzen, hier kann man einstellen, dass die Zeilennummer angezeigt wird.

4. Sich bei schwierigen Fragen von einem Freund helfen lassen

Als erstes stelle ich meine Frage in Google. Wenn die Frage konkret genug gestellt ist, finde ich auch ein Forum, das sich damit schon mal befasst hat. Man kann auch selbst im Forum seine Frage stellen und bekommt auch meist eine Antwort.

Man stellt die Frage an einen persönlichen Freund, evtl. unter Benützung des Teamviewers. Dies habe ich schon mehrfach getan, entweder als Fragesteller oder als Antwortgeber mit Teamviewer.

Teamviewer: <https://heinz-wember.de/dokus/html/teamviewer.html>

Dokumentation zu SFTP und Teamviewer

Zugang zum Server xxx.de mit sftp (secure ftp)

1. sftp-Client: WinSCP Download bei <https://heise.de/software/download/winscp/5962>
2. Installieren des Programms: Ausführen der heruntergeladenen Datei winscp430setup.exe
3. Aufruf des Programms WINSCP
 - 3.1. Neuerstellung eines Zugangs: in vorname-nachname.de , in u99999999
 - 3.2. Aufruf eines bestehenden Logins: linkes Fenster im rechten Fenster die gewünschte Verbindung markieren
4. Einstellungen (von mir bevorzugt) : linkes Fenster einstellen, wo normalerweise die zu übertragenden Dateien stehen z.B. D:\homepage
linkes Fenster: bei Commander markieren

Zum Support über Netzwerk Programm Teamviewer

1. Download bei <https://teamviewer.com/de/>
2. Installieren der heruntergeladenen Datei: TeamViewer_Setup_de.exe
3. Aktivitäten für denjenigen, der den Zugriff auf seinen PC zulässt:
Aufruf von Teamviewer
Auf der linken Seite des Fensters stehen die eigenen Zugriffsdaten:
Dem Supportmann die ID und das Kennwort über Telefon mitteilen (Das Passwort ist nach Beendigung der Sitzung verfallen, bei einer neuen Sitzung wird auf jeden Fall ein neues Passwort vergeben)
4. Aktivitäten für den Supportmann
Aufruf: linke Seite eigene Daten Id und Kennwort
Rechte Seite die Partner-ID eingeben und Fernsteuerung auswählen. Damit hat man auf den Partner-PC die volle Zugriffsmöglichkeit

Quellen und Literatur

Quellen:

Staatsarchiv Augsburg:

<https://heinz-wember.de/MySQL-Kriegshaber/katbandn.php>

Bistumsarchiv Augsburg

<https://heinz-wember.de/MySQL-Kriegshaber/VerzeichnisKBKriegshaber-Oberhausen.htm>

Bild der Quellen des Bistumsarchivs

Quellen: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/baa.jpg>

Literatur:

1und1: <http://hosting.1und1.com/?kww=2076168>

html-Handbuch: <http://de.selfhtml.org/>

PHP-Manual; <http://www.php.net/manual/de/>

PHP4-Forum PHP-Kurs: <http://www.php4-forum.de/>

PHP Claudia Schattenbaum: <http://www.schattenbaum.net/php/>

PHP4-Forum PHP-Kurs: <http://hosting.1und1.com/?kww=2076168>

MySQL Reference 4.0, 4.1 <http://dev.mysql.com/doc/refman/4.1/en/>

MySQL Reference 5.1: <http://hosting.1und1.com/?kww=2076168>

PHP und MySQL: <http://reeg.junetz.de/DSP/main.html>

PHP und MySQL faq: <http://mysql-faq.sourceforge.net/>

MySQL-admin: <http://www.mysql-admin.org/>

Apache-Wamp: <http://www.wampserver.com/en/>

XAMPP: <http://www.apachefriends.org/en/xampp.html>

MySQL-Handbuch: <http://www.little-idiot.de/mysql/>

MySQL Seminar: <http://www.tbee.de/mysql/>

Homepage-Forum: <http://www.homepage-forum.de/showthread.php?t=11080>

PHP und MySQL auf der Homepage Preis:: Barrierefreies Webdesign: <http://www.jchanke.de/knowware-hefte/29-php-und-mysql-auf-der-homepage>

Webdatenbanken mit PHP und MySQL: <http://www.bilke.de/mysql/index.htm>

PHP und MySQL Praxisbuch für Kids, Johann-Christian Hanke, Heidelberg 2006

PHP5 MySQL5 Matthias Kannengiesser, 2009 Franzis Verlag Poing

Tools, mit denen ich dieses Buch geschrieben habe

Vielleicht interessiert es den einen oder anderen Leser, wie ich dieses Buch geschrieben habe. Die meisten schreiben wohl das Konzept ihrer Veröffentlichung mit einem Textprogramm, Word, Open Office oder sonst einem ähnlichen Programm. Ich habe eine gewisse Abneigung gegen diese Programme. Vielleicht liegt es daran, dass ich 1980 ein Textverarbeitungsprogramm mittels dem Betriebssystem IMOS und der Programmiersprache Cobol schreiben musste, (gedrängt nach diesem Projekt hatte ich mich damals nicht). Es war auch nach einem Jahr fertig und das Schreibbüro in der Fabrik der NCR in Augsburg hat mehrere Jahre dieses Programm zum Schreiben der umfangreichen Dokumentationen für unsere Produkte auch benutzt bis eben etwas Besseres da war, damals ein NCR-PC mit Betriebssystem DOS und einem Textverarbeitungssystem (damals noch nicht Word).

Ich mache das ganz anders, ohne zu behaupten, dass es der bessere Weg wäre. Kurz zusammengefasst, ich habe es mit den Tools geschrieben, die ich in diesem Buch vorgestellt habe. Als erstes war das Buch als Internet-Buch fertig, d.h. wie gewohnt im Internet einzelne Dateien in MySQL-DB erfasst, gestartet vom Programm index.html, das das Inhaltsverzeichnis aufruft. Hierüber kann man die Kapitel aufrufen. Soweit es Abbildungen sind, wurden diese entweder als reine Textdateien mit WinVi32 erstellt (Endung txt) oder als jpg-Dateien, die das Programm Photoshop erstellt hat. Bildschirmseiten kann man auch vom Bildschirm "fotografieren", also die Tasten Shift und Druck und den dann gespeicherten Inhalt im Programm Photoshop einfügen und anpassen. Diese txt- oder jpg-Dateien kann man mittels Links aufrufen. Damit ist das Buch eigentlich schon fertig, wenn man es nicht als wirklich gedrucktes Buch herausgeben will.

Wie entsteht daraus ein gedrucktes Buch?

Nachdem alle Texte in Tabellen vom Datentyp alphanumerisch in der Datenbank MySQL stehen, kann man diese Tabellen ohne weiteres in das Programm Indesign kopieren. Mit den Links und den dahinter stehenden text- oder jpg-Dateien muss man es anders machen. Alle html-Steuerzeichen muss man jetzt aber entfernen. Dies geschieht dadurch, dass man die Links mit einem Browser (z.B. firefox) in einem neuen Fenster aufruft und dann den gesamten Text markiert und kopiert.

Indesign gestattet das Einfügen dieses Textes aus dem Zwischenspeicher, auch von txt- und jpg-Dateien. Dabei ist das Programm sehr vornehm, was das Anpassen an das Druckformat betrifft. Sollte sich jemand interessieren, wie Indesign und Photoshop anzuwenden sind, auf meiner Homepage stehen ein paar Beispiele. Am Ende wird die Indesign-Datei als pdf-Datei exportiert, die eine Druckerei ohne Probleme drucken und binden kann.

Nachwort

Dieses Buch, das Interesse der Geisteswissenschaftler für die Möglichkeiten, die die modernen Medien bieten können, wecken sollte, ist natürlich nicht der eigentliche Zweck meiner Forschungen zur Geschichte meines Wohnortes. Es sollte vielleicht neugierig machen, was es zur Geschichte von Kriegshaber noch zu sagen gibt, nachdem es ja die Dissertation von Luis Dürrwanger aus dem Jahre 1935 schon gibt.

Luis Dürrwanger hat in seinem Vorwort in seiner Dissertations-Schrift bereits gesagt, dies sei nur der erste Teil. Den zweiten und dritten Teil konnte er erst wegen der politischen Verhältnisse in Deutschland nicht veröffentlichen. Nach dem Krieg suchte er vergeblich nach Sponsoren für die weiteren Teile. Als sich die wirtschaftliche Situation wieder gebessert hatte, ist er 1959 gestorben. Seitdem liegt sein Nachlass zu diesem Thema im Stadtarchiv Augsburg.

Nachdem ich seit 20 Jahren als Gasthörer an der Uni Augsburg Geschichte studiere, habe ich mich auch mit dem Thema Kriegshaber befasst. Jetzt habe ich das Problem, wie kann ich ein Buch kostengünstig bzw. kostendeckend veröffentlichen? Durch den Verein Familienkunde Schwaben lernte ich das Staatsarchiv Augsburg kennen und die Möglichkeiten, die die historischen Kataster für die Familienkunde bieten. Dieses Thema war dann so umfangreich, dass wieder nicht an eine Veröffentlichung in Buchform zu denken war. Wer liest schon ein Buch mit mehreren tausend Seiten. Aus diesem Dilemma heraus ist mein Vorhaben entstanden, die vielen Quellen nicht in Buchform, sondern im Internet zu veröffentlichen.

Ein Problem ist leider zur Zeit noch offen. Nachdem die Version php 5.5 einwandfrei im Internet und auf meinem PC mittels XAMPP lief, damals lief auf dem Laptop Windows 7 mit 32 bit, habe ich mich entschlossen, die php-Version auf 7.2 zu aktualisieren. Im Internet bei meinem Provider 1und1 läuft alles mit der neuen php-Version 7.2 bestens. Zur Zeit (Juni 2019) ist das Problem mit der falschen Darstellung der Umlaute auf der Windows-Version nicht gelöst, zumindest habe ich es nicht geschafft. Es stehen ausgesprochen viele Rezepte im Internet, wie man das schaffen sollte, leider ist es mir nicht gelungen. Nachdem ich meine Forschungen in den Archiven abgeschlossen habe, ist es auch für mich nicht mehr vordringlich, dieses Problem zu lösen. Ich werde von Zeit zu Zeit nach neueren Versionen von XAMPP schauen, ob nun eine einfache Lösung für dieses noch offene Problem gibt. Mein alter Laptop, auf dem Windows7 mit 32 bit-Version lief, ist an Altersschwäche gestorben. Auf meinem neuen Laptop mit Windows10 64 bit habe ich es vorerst nicht geschafft, die 64-bit-Version von XAMPP so zu installieren, dass dies mit den Länder spezifischen Zeichen (ä ö ü ß) ohne weiteres läuft, ohne dass ich die Texte mit einem anderen Languagecode neu schreiben muss.

Überhaupt ist es immer kritisch, wenn man Texte mit Länder spezifischen Zeichen von Windows- zu Linux-Plattformen transferiert. Viele Applikationen, die in USA geschrieben wurden, kümmern sich grundsätzlich nicht um die Länderzeichen in der Welt. Der einfache Weg, den ich verfolge, ich verwende z.B. bei Dateinamen nur Zeichen, die das amerikanische Keyboard kennt, außerdem vermeide ich Leerzeichen im Dateinamen. Dann kann man sicher sein, dass dies die US-Applikationen unter Linux richtig interpretieren.

Anhang: Liste der aufgeführten Beispiele und Abbildungen, mit php7.2

die Beispiel-Applikation: <https://heinz-wember.de/MySQL-Kriegshaber/index.html>

Dokumentationen: <https://heinz-wember.de/dokus/index.html>

Änderungsbeispiele von php 5.5 zu php 7.2: <https://heinz-wember.de/MySQL-kriegshaber/buchdoku/Do-kuAendgRel5zuRel7.txt>

Änderung einer Tabelle, die kein Feld mit auto-increment hat: <https://heinz-wember.de/MySQL-kriegshaber/buchdoku/Feldincrement.txt>

Umlautedarstellung: Richtige Darstellung der Umlaute unter php 7.2

Fotodokumentation Version relative Spaltenbreite: <https://heinz-wember.de/MySQL-rel-FotoDoku/index.html>

Fotodokumentation Version fixe Spaltenbreite: <https://heinz-wember.de/MySQL-fix-FotoDoku/index.html>

Foto-Dokumentation: Download der Programme <https://heinz-wember.de/index-fotodoku.html>

Datenbankverknüpfungen: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/dokutabellenverkn.php>

Programmierung: <https://heinz-wember.de/MySQL-Kriegshaber/dokuprogrammierung.php>

Programmierung Massendatenänderung: <https://heinz-wember.de/MySQL-Kriegshaber/dokudbmasse-naendg.php>

Doku Update MySQL: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/dokupdb.php>

DB-Definition: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/db-definition.txt>

Create Tabelle dokumentation: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/crtabdokumentation.txt>

Create Tabelle hochzeiten: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/crtabhochzeiten.txt>

Create Tabelle kreisentfernung: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/crtabkreisentfernung.txt>

Create Tabelle navigation: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/crtabnavigation.txt>

Create Tabelle ortkreis: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/crtabortkreis.txt>

Create Tabelle uebersicht: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/crtabuebersicht.txt>

Select-Beispiele: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/dokuselect.txt>

Format-Datei: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/FORMATE.CSS>

Programm gestaltung.php: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/gestaltung.txt>

Programm hauseval1918.php: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/hauseval1918.txt>

Programm index.html: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/buchdoku/index.txt>

Programm katband.php: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/katband.txt>

Programm mrimkrse001bis020.php: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/mrimkrse-001bis020.txt>

Programm navigation.php: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/navigation.txt>

Programm ngbdetart.php: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/ngbdetart.txt>

Programm progdatum2.php: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/progdatum2.txt>

Programm suchnamen.php: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/terminedatwozuk.txt>
Programm top.php: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/top.txt>
Programm uebersicht.php: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/uebersicht.txt>
Programm DB öffnen: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/zugriff-dbkriegshaber-doku.txt>
Ausgabe Datum formatiert: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/Ausgabe-Datum.jpg>
Ausgabe Datum normal: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/Ausgabe-Datum2.jpg>
Ausgabe Apotheker: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/Auswahl-Apothet.jpg>
Ausgabe Bevölkerungsstatistik: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/BevStatistikErg.jpg>
Ausgabe Katband1: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/KatBand1.jpg>
Ausgabe Katband2: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/KatBand2.jpg>
Ausgabe Quellen Bistumsarchiv: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/QBbistumsAA.jpg>
Ausgabe StAAFindbuchblatt: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/StAAFindbuchblatt.jpg>
Ausgabe Navigation: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/Tabnavigation.jpg>
Ausgabe Textgestaltung: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/textgestaltung.jpg>
FotoDoku connect auf die MySQL-DB: php rel. 7.2 <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/zugriff-db-fotodoku.inc-php.txt>
FotoDoku fixe Spaltenbreite index.html php rel. 7.2 <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/index.html-rel.txt>
FotoDoku relative Spaltenbreite navigation php rel. 7.2 <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/navigation01-php.txt>
FotoDoku relative Spaltenbreite php rel. 7.2 <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/fotos-nachablageort-php.txt>
FotoDoku relative Spaltenbreite php Rel. 7.2 <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/motivauswahl.txt>
Ausgabemotivauswahl-rel: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/Ausgabemorivauswahl-rel.jpg>
FotoDoku fixe Spaltenbreite index.html php Rel. 7.2 <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/index.html-fix.txt>
FotoDoku fixe Spaltenbreite php Rel. 7.2 <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/fotos-nachgruppeundjahrfix-php.txt>
FotoDoku fixe Spaltenbreite php Rel. 7.2 <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/persauswahlfix.txt>
Ausgabepersonauswahlfix: <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/AusgabePersonauswahl-fix.jpg>
Verwendung von html-Steuerzeichen in Tabellen <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/pflegetabellen.txt>
Besonderheiten mit Frameset <https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/index.frame.html>

Änderungsstand buchanhang: 25-Apr-2015 / Upd 13-Jul-2019
Heinz Wember

Anhang: Dateien und Skripte, die mit dem unten stehenden Aufruf, heruntergeladen werden können:

<https://heinz-wember.de/MySQL-Kriegshaber/buchdoku/buchanhang.php>

Nachdem abzusehen ist, dass meine Homepage mich eventuell nicht überleben wird, kommen in dieser 2. Auflage alle die Programme, die es zu Zeit noch online gibt, hier als Druck heraus.

(25-Mar-2020)

Beispiele für häufiges Vorkommen veralteter php Befehle in Rel. 5.5 und entsprechender Befehl in Rel. 7.2 bei meinen Programmen

Connect DB

alt:
\$db = mysql_connect (MYSQL_HOST, MYSQL_BENUTZER, MYSQL_KENNWORT);
if (\$db)
{
}
else
{
echo „Datenbank db9999999 (abitur) down, bitte benachrichtigen Sie Post@Heinz-Wember.de“;
}
neu:
\$db = mysqli_connect (MYSQL_HOST, MYSQL_BENUTZER, MYSQL_KENNWORT, MYSQL_DATENBANK);

if (\$db)
{
}
else
{
echo „Fehler: konnte nicht mit MySQL verbinden.“ . PHP_EOL;
echo „Debug-Fehlernummer: „ . mysqli_connect_errno() . PHP_EOL;
echo „Debug-Fehlermeldung: „ . mysqli_connect_error() . PHP_EOL;
exit;
}

Open db

Befehl nach include „../db/zugriff-db-fotodoku.inc.php“;
alt: mysql_select_db(MYSQL_DATENBANK) or die(mysql_error());
neu: entfällt, da bereits im neuen Unterprogramm zugriff-db-fotodoku.inc.php enthalten

Query:

alt: \$ergebnis = mysql_query(\$sql);
neu: \$ergebnis = mysqli_query(\$db, \$sql);

Schleife für die Ausgabe der Sätze

alt: while(\$row = mysql_fetch_object(\$ergebnis))
neu: while(\$row = \$ergebnis->fetch_assoc())

Ausgabe der DB-Felder:

alt: echo „<td style=width: 10%;> \$row->zeit</td>“;
neu: echo „<td style=width: 10%;> \$row[zeit]</td>“;

Schliessen der Datenbank:

alt: Befehl fehlt komplett
neu: mysqli_close(\$db);

Änderungsstand: 06-Feb-2019

Heinz Wember

Aufgabe: in einer Tabelle, die das Feld id hat, das noch nicht mit increment definiert ist, einen aufsteigenden Wert ins Feld id setzen:

```
SET @num := 0;  
UPDATE your_table SET id = @num := (@num+1);  
ALTER TABLE your_table AUTO_INCREMENT =1;
```

vorher prüfen, ob evt. ein Satz mit id=0 vorhanden ist, wenn ja, diesen die höchste Nummer geben.
anschließend die Strukturänderung durchführen:
Struktur Feld id Standard keine Feld a_i anhängen
bei der Übersicht Struktur erscheint nun AUTO_INCREMENT in der Spalte EXTRA

Heinz Wember
25-Feb-2019

Dokumentation über die DB-Tabellenverknüpfungen

Wie allgemein bekannt, ist der große Vorteil einer Datenbank, dass man Tabellen verknüpfen kann. Dies ist der große Unterschied zu einfachen Tabellenprogrammen wie z.B. Excel. Grundsätzlich ist aber eine DB-Tabelle ähnlich aufgebaut wie eine Excel-Tabelle. Was nun den großen Unterschied ausmacht, sind die möglichen Verknüpfungen. Verknüpfungen macht man immer dann, wenn im Ergebnisfeld einer Auswertungen alle Sätze gleiche Felder haben, wie z.B. alle Häuser haben eine Adresse (Straßenname und Hausnummer). Jedoch gibt es in jeder Straße meist viele Häuser. Damit man nun nicht bei jedem Haussatz den langen Straßennamen eingeben muss, löst man dies dadurch, dass man eine weitere Tabelle anlegt, die nur den Straßenschlüssel und den Straßennamen enthält. In die Tabelle für die Häuser speichert man nur den Straßenschlüssel, der im allgemeinen wesentlich kürzer ist. Damit ist aber auch verbunden, dass alle Häuser in einer Straße eindeutig denselben Straßennamen haben. Entweder ist der Straßename bei allen relevanten Sätzen richtig oder bei allen falsch. Wenn man dies mit einer Excel-Tabelle lösen wollte, muss man für jedes Haus den langen Straßennamen eingeben.

Ein anderes, komplexeres Beispiel.

Man will z.B. die Archivordner katalogisieren. Ich nehme gleich das konkrete Beispiel: die Katasterbände im Staatsarchiv Augsburg. Es ist klar, dass es zu einem Ort viele verschiedene Katastertypen gibt, z.B. einen Kataster vom Jahre 1867 und einen vom Jahre 1892. Was die beiden Kataster unterscheidet ist zumindest der Name, einmal Kataster 1867 und dann Kataster 1892. Gleich ist bei beiden Typen der Ort, in unserem Beispiel Kriegshaber (Rentamt Augsburg-Land). Nun gibt es aber mehrere Bände dieser Kataster, in dem einem Band sind z.B. alle Häuser von 1 bis 50 in dem anderen von 51 bis Ende verzeichnet. Also hat man zwei Bänder eines Katastertypen. Nun kommt noch hinzu, dass der Katastertyp einen relativ langen Namen hat, eine Abkürzung hierfür würde die Datenerfassung leichter machen, also wir erfassen bei den Bänden nur k1867 bzw. k1892. Den ausführlichen Namen legen wir in eine neue Tabelle ab, die wir z.B. abk (Abkürzung) nennen.

Man hat nun mehrere Abhängigkeiten (Verknüpfungen). Die Bände gehören verschiedenen Katastertypen an, haben aber dieselben Kopfdaten (Ort und Rentamt). Ebenso die Katastertypen, es gibt mehrere Typen, die aber auch dieselben Kopfdaten haben. Schließlich verwendet man bei den Katastertypen und auch bei den Katasterbänden die Abkürzungen für die Katastertypen.

```
LEFT JOIN katkopf ON katkopf.raort = katband.raort  
LEFT JOIN kattypn ON katband.typ = kattyp.typ  
LEFT JOIN katabk ON kattyp.abk = katabk.abk
```

Schwieriger ist die Systematik, wenn man unterschiedliche Orte erfassen will. Hier ist der Typ nicht nur vom Feld abk abhängig, sondern auch vom Ort (Feld raort). Erstens hat nicht jeder Ort die gleiche Anzahl von verschiedenen Katastertypen, sondern auch die Jahrbereiche, die in dem speziellen Kataster erfasst sind, sind unterschiedlich. Deshalb ist das Verknüpfungsfeld typ von katband zu kattyp eine Kombination von der Felder raort und abk. Der Left Join sieht dann so aus:

```
LEFT JOIN katkopf ON katkopf.raort = katbandn.raort
```

```
LEFT JOIN kattypn ON katbandn.typ = kattypn.typ
```

```
LEFT JOIN katabk ON kattypn.abk = katabk.abk
```

Bei der Datenerfassung für die Katasterbände sollte man top down vorgehen:

erst erfasst man alle Orte (also Rentamt und Ort)

dann alle möglichen Abkürzungen: also in unserm Beispiel gr1, gr2 und gr3

nun kann man alle Katastertypen erfassen, also z.B. gr1, gr2 und gr3, (bzw. für das allgemeine zweite Beispiel askrigr1, askrigr2, askrigr3) die natürlich noch weitere Felder beinhalten (Gruppe, Jahrbereiche), schließlich nun die Erfassung der Bände, hier werden die Verknüpfungsschlüssel raort und abk (bzw. typ für das zweite Beispiel) verwendet, die man bereits früher erfasst hat.

Nun kommt es darauf an, mit welchem Programm man die Tabellen füllt, nimmt man ein spezielles Programm für jede Tabelle, dann kann man gleich die Verknüpfungen prüfen, ich kann also keinen Band für einen Ort eingeben, der vorher nicht erfasst wurde. Ich habe mir diese Erfassungsprogramme samt und sonders gespart und erfasse mit dem Administratorprogramm (adminmysql). Hier ist es im Gegensatz zu anderen Datenbanken (z.B. Oracle) nicht möglich, bei der Struktur auch gleich die Verknüpfungsregeln zu prüfen. Wenn ich also einen Ort vergessen habe, kommt bei der verknüpften Liste nun kein Ortsname und Rentamtsname, analog bei den Abkürzungen. Dies ist aber augenscheinlich, man sieht sofort auf Grund des Verknüpfungsbegriffes was noch in anderen Tabellen zu ergänzen ist.

Nun ein Beispiel, wo es nicht so einfach ist, fehlende Verknüpfungen sofort zu erkennen. Hier also die Verknüpfung der Häusersätze mit den Straßennamen. Hier ist es so (bei dem hier verwendeten JOIN), dass nun nur diejenigen Häuser angezeigt werden, für die ein Satz mit dem Straßennamen vorhanden ist. Wenn nun das eine oder andere Haus in meiner Liste fehlt, fällt das natürlich nicht auf, man hakt ja nicht die ganze Liste ab. Daher habe ich für diesen Zweck ein separates Prüfprogramm geschrieben.

Das Programm habe ich haeusereval1913.php genannt, die Source-Daten dieses Programms sind folgendermaßen:

```
$abfrage1 = „SELECT haeuser.ID, haeuser.hausnr1913, haeuser.schluessel1913, haeuser.besitzer1913
```

```
FROM haeuser
```

```
LEFT JOIN statamtstrassen ON haeuser.schluessel1913 = statamtstrassen.schluessel
```

```
WHERE statamtstrassen.schluessel is NULL“;
```

```
$ergebnis1 = mysql_query($abfrage1);
```

```
$ergebnis = mysql_query($abfrage);
```

```
while($row = mysql_fetch_object($ergebnis)) {
```

```
echo „ \nID= $row->ID“;
```

```
echo „ hausnralt= $row->hausnralt“;
```

```
echo „ schluessel1913= $row->schluessel1913“;
```

```
echo „ besitzer1913= $row->besitzer1913“; }
```

```
?>
```

Das Ergebnis dieser Prüfung ist folgende: ID= 620; hausnr1913= Z9999; schluessel1913= 999999; besitzer1913 = ZZ-Änderungsstand (haeuser): 17-Jul-2012 06:00

Analoge Prüfungen für folgende Verknüpfungen:

ON haeuser.schluesel1918 = statamtstrassen.schluesel und

ON haeuser.strassenr = statamtstrassen.schluesel

Ich werde darauf hingewiesen, dass entweder die Straße 999999 in der Tabelle strassen1 fehlt oder ich habe mich beim Eingeben des Satzes mit der ID=620 in der Tabelle haeuser vertippt. Wenn ich die Programme Hausnummer alt zu Hausnummer neu, oder Hausnummer neu zu Hausnummer alt aufrufe, bemerkt man diesen Fehler überhaupt nicht, weil ja nur diejenigen Sätze angezeigt werden, bei denen die strassenr in beiden Tabellen vorhanden ist.

Siehe hierzu im Manual MySQL Cookbook das Kapitel 12 Using Multiple Tables.

Änderungsstand dokutabellenverkn: 22-Okt-2014 Upd 19-Dez-2018

Heinz Wember

Zur Programmierung dieses Systems

Die grundsätzlichen Überlegungen für diese Programmierungsart (Apache-Server und MySQL-Server) habe ich schon beschrieben, siehe Seite 6.

Das System kann man entweder online bei einem Homepage-Provider (ich verwende 1und1.de) oder man kann es auch lokal auf einem Windows-PC (oder auch Linux-Maschine) machen, ich verwende beides, die Datenerfassung mache ich im Archiv lokal auf dem Laptop, zu Hause kopiere ich die MySQL-DB ins Internet.

Nun das Startprogramm, es heißt index.html und öffnet drei Fenster: top, navigation und content, das das Hauptfenster ist.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" „http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd“>
<html>
<head>
<style type="text/css">
  @import url(„FORMATE.CSS“) screen;
</style>
<title>Straßen, Häuser und Kataster von Kriegshaber</title>
<meta name="author" content="Heinz Wember Augsburg">
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>
<frameset rows="130,*">
<!-- Dadurch ergeben sich zwei Frame-Fenster, deren Inhalt hier bestimmt wird.
  Das erste Frame-Fenster wird normal definiert, anstelle des zweiten
  folgt jedoch das folgende untergeordnete Frameset: -->
<frame name="top" src="top.php" scrolling="no" frameborder="no">
<frameset cols="15%,85%"rows=1200 >
<frame name="navigation" src="navigation.php" scrolling="auto" frameborder="no" >
<frame name="content" src="strassennamen.php „, scrolling="auto" frameborder="no">
</frameset>
</frameset>
<body>
Datei index.html

</body></noframes></body></html>
```

Hier das Programm top.php:

```
<!DOCTYPE html PUBLIC „-//W3C//DTD HTML 4.01 Transitional//EN“>
<html xmlns=“http://www.w3.org/1999/xhtml“ xml:lang=“de“>
<head>
  <style type=“text/css“>
    @import url(„FORMATE.CSS“) screen;
  </style>
  <meta name=“author“ content=“Heinz Wember Augsburg, php7, 04-Mar-2019“>
</head>
<body>
<!-- web:picture:start --><img src=“start1.jpg“ border=“0“ width=“1600“ alt=“Bild“ title=“Bild“
height=“130“/><!-- web:picture:stop -->
</body>
</html>
```

Hier das Programm navigation.php:

```
<!DOCTYPE html PUBLIC „-//W3C//DTD HTML 4.01 Transitional//EN“>
<html xmlns=“http://www.w3.org/1999/xhtml“ xml:lang=“de“>
<head>
  <style type=“text/css“>
    @import url(„FORMATE.CSS“) screen;
  </style>
  <meta name=“author“ content=“Heinz Wember Augsburg, php7, 16-Feb-2019“>
</head>

<left>
<small>
<table width=15%; text-align: left; margin-left: auto; margin-right: auto;“ border=“1“ cellpadding=“2“
cellspacing=“2“>
  <tbody>
<?php
  include „../db/zugriff-dbkriegshaber.inc.php“;
  $abfrage = „SELECT gruppe , ugruppe , titel
    FROM navigation
    WHERE gruppe = ‘00‘ and aktiv = ‘j‘
    ORDER BY ugruppe“;
  $ergebnis = mysqli_query($db, $abfrage);
  while($row = $ergebnis->fetch_assoc())
  {
    echo „<tr></b> $row[titel] </tr> <br>“;
  }
  mysqli_close($db);

?>
  </tbody>
</table>
</small>
</body>
</html>
```

Hier das Einleitungsprogramm:

```
<!DOCTYPE html PUBLIC „-//W3C//DTD HTML 4.01 Transitional//EN“>
<html xmlns=“http://www.w3.org/1999/xhtml“ xml:lang=“de“>
<head>
  <style type=“text/css“>
    @import url(„FORMATE.CSS“) screen;
  </style>
  <meta name=“author“ content=“Heinz Wember Augsburg, php7, 04-Mar-2019“>
  <title>Übersicht Straßennamen in Kriegshaber</title>
</head>
<body>
<br><center>
<h1>Übersicht Straßennamen in Kriegshaber</h1>
<br>
<?php
  define(„UNLOCK“,1);
  include „../db/zugriff-dbkriegshaber.inc.php“;
  $abfrage = „SELECT text from strassennamen“;
  $ergebnis = mysqli_query($db, $abfrage);
  while($row = $ergebnis->fetch_assoc())
  {
    echo „,$row[text]“;
  }
  mysqli_close($db);
?>

<hr>
<center>
<a target=“_blank“ href=“index.html“>Übersicht</a>
</center>
<hr>
<small>Änderungsstand Programm: 04-Mar-2019<br>
<a href=“../Impressum.html“>Heinz Wember</a></small>
</body>
</html>
```

Von der Technik spielt es keine Rolle, welches Programm man hier auswählt. Sinnvollerweise nimmt man ein Programm, das in das System einführt, hier in diesem Fall das Programm strassennamen.php

Es ist wichtig, dass jede Tabelle ein Feld, bei mir id genannt, enthält, das erstens einen unique-Schlüssel hat und auch mit autoincrement definiert ist. Damit ist ein sicheres Updaten der Tabelle möglich. Hat man z.B. dieses Feld vergessen oder man ändert ein fremdes System, so kann man dies auch nachher machen.

Aufgabe: in einer Tabelle, die das Feld id hat, das noch nicht mit increment definiert ist, einen aufsteigenden Wert ins Feld id setzen:

```
SET @num := 0;
UPDATE your_table SET id = @num := (@num+1);
ALTER TABLE your_table AUTO_INCREMENT =1;
```

vorher prüfen, ob evt. ein Satz mit id=0 vorhanden ist, wenn ja, diesem die höchste Nummer geben.
anschließend die Strukturänderung durchführen:

Struktur Feld id im Standard=keine und Feld a_i anhaken

bei der Übersicht Struktur erscheint nun AUTO_INCREMENT in der Spalte EXTRA

Zur richtigen Darstellung der Umlaute bei MySQL und php

Hier diese Anweisung gilt z.B. wenn man bei ionos das System generiert hat.

Antwort des Betreuers bei ionos:

ich habe in dem Verzeichnis "/htdocs/MySQL-FotoDoku" eine php.ini angelegt.

In der Datei ist "default_charset=iso-8859-1" hinterlegt.

Dadurch werden die Umlaute z. B. auf <https://heinz-wember.de/MySQL-FotoDoku/navigation.php> korrekt angezeigt.

Wenn Sie die Einstellung auf alle Ordner übertragen möchten, legen Sie bitte im Hauptverzeichnis eine ".user.ini"

Datei an mit "default_charset=iso-8859-1". Dadurch wird die Einstellung auf alle Unterverzeichnisse vererbt.

Massendatenänderung bei einer Tabelle durchführen:

Dieses Beispiel zeigt, wie man eine Massenänderung in einer SQL-Datenbanktabelle durchführen kann.

1. Die betreffende Tabelle, bei der die Massenänderungen durchgeführt werden soll, exportieren und auf dem PC speichern.
2. Diejenige Tabelle, die man gerade exportiert hat, umbenennen z.B. in tabellealt. Dies ist sinnvoll, um später kontrollieren zu können, ob die Massenänderung gelungen ist.
3. Die exportierte Datei aus 1. mit einem Editor z.B. vi oder Äquivalent von Windows (z.B. winvi32.exe) aufrufen und hier nun die Änderungen mit der bekannten Funktion Suchen/Ändern durchführen, z.B. house in Haus ändern. Es ist darauf zu achten, dass diese Änderung alle Felder der Tabelle betrifft.
4. Die so modifizierte Datei wieder in MySQL importieren.
5. Kontrollieren der Massenänderung durch Vergleich der alten und neuen Tabelle.

Achtung: Dies nur tabellenweise durchführen, nicht für die ganze Datenbank.

Definition der Datenbank kriegshaber

```
-- phpMyAdmin SQL Dump
-- version 4.6.6deb4
-- https://www.phpmyadmin.net/
--
-- Host: infongd-eu6168.rtr.schlund.de
-- Erstellungszeit: 10. Jul 2019 um 13:51
-- Server-Version: 5.5.60-0+deb7u1
-- PHP-Version: 7.0.33-0+deb9u3

SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
SET time_zone = "+00:00";
/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Datenbank: `db288973340`
--
CREATE DATABASE IF NOT EXISTS `db288973340` DEFAULT CHARACTER SET latin1 COLLATE
latin1_swedish_ci;
USE `db288973340`;

-----
--
-- Tabellenstruktur für Tabelle `BrowserEinst`
--

CREATE TABLE IF NOT EXISTS `BrowserEinst` (
  `id` int(4) NOT NULL AUTO_INCREMENT,
  `text` text COLLATE latin1_german1_ci NOT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIME-
STAMP,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Brower
Einstellungen';

-----
--
-- Tabellenstruktur für Tabelle `buchschrBuch`
--

CREATE TABLE IF NOT EXISTS `buchschrBuch` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `text` text COLLATE latin1_german1_ci NOT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIME-
STAMP,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci
COMMENT='Dokumentation MySQL und Geschichte';

-----
```

```
--  
-- Tabellenstruktur für Tabelle `gefaelle1824`  
--
```

```
CREATE TABLE IF NOT EXISTS `gefaelle1824` (  
  `id` int(4) NOT NULL AUTO_INCREMENT,  
  `stgem` varchar(10) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL DEFAULT  
,askri`,  
  `hausnr` varchar(10) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,  
  `plannr` varchar(10) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,  
  `besitzer` varchar(255) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,  
  `besitzung` varchar(40) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL DEFAULT  
,vom Leerhaus`,  
  `art` varchar(10) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL DEFAULT  
,Grundzins`,  
  `betragfl` decimal(10,0) DEFAULT ,0`,  
  `betragkr` decimal(5,0) DEFAULT ,0`,  
  `betraghl` decimal(5,0) DEFAULT ,0`,  
  `bemerkung` text CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,  
  `quelle` varchar(30) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL DEFAULT  
,Gefällbuch 15, Seite 15`,  
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIME-  
STAMP,  
  
  PRIMARY KEY (`id`),  
  KEY `stgem` (`stgem`)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COMMENT='Gefälle 1824';
```

```
-----  
  
CREATE TABLE IF NOT EXISTS `grundbsachr` (  
  `id` int(10) NOT NULL AUTO_INCREMENT,  
  `raort` varchar(100) COLLATE latin1_german1_ci NOT NULL DEFAULT ,askri`,  
  `flurnr` varchar(100) COLLATE latin1_german1_ci NOT NULL,  
  `bandseite` varchar(100) COLLATE latin1_german1_ci NOT NULL,  
  `bemerkung` varchar(100) COLLATE latin1_german1_ci NOT NULL,  
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIME-  
STAMP,  
  PRIMARY KEY (`id`),  
  KEY `raortabk` (`raort`)  
  ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Grundbuch  
Sachregister';  
  
-----
```

```

-----
-- Tabellenstruktur für Tabelle `haeuser`
--
CREATE TABLE IF NOT EXISTS `haeuser` (
  `ID` int(4) NOT NULL AUTO_INCREMENT,
  `plannr` varchar(100) COLLATE latin1_german1_ci DEFAULT NULL,
  `hausnr1867` varchar(10) COLLATE latin1_german1_ci DEFAULT NULL,
  `besitzer1867` text COLLATE latin1_german1_ci,
  `hausnr1892` varchar(10) COLLATE latin1_german1_ci DEFAULT NULL,
  `besitzer1892` text COLLATE latin1_german1_ci,
  `strassenr1913` int(6) DEFAULT NULL,
  `hausnr1913` varchar(20) COLLATE latin1_german1_ci DEFAULT NULL,
  `besitzer1913` text COLLATE latin1_german1_ci,
  `name1913buch` varchar(60) COLLATE latin1_german1_ci DEFAULT NULL,
  `kzdruck` varchar(1) COLLATE latin1_german1_ci NOT NULL DEFAULT 'j' COMMENT 'Kennzeichen
Satz drucken für Buch',
  `strassenrheute` varchar(6) COLLATE latin1_german1_ci DEFAULT NULL,
  `hausnrheute` varchar(100) COLLATE latin1_german1_ci DEFAULT NULL,
  `strassenr1918` int(6) DEFAULT NULL,
  `hausnr1918` varchar(100) COLLATE latin1_german1_ci DEFAULT NULL,
  `hausnrumschr` varchar(10) COLLATE latin1_german1_ci DEFAULT NULL,
  `besitzerumschr` text COLLATE latin1_german1_ci,
  `ksumschr` text COLLATE latin1_german1_ci,
  `repermessv` text COLLATE latin1_german1_ci,
  `hausnr1854` varchar(10) COLLATE latin1_german1_ci DEFAULT NULL,
  `besitzer1854` text COLLATE latin1_german1_ci,
  `hausnr1840` varchar(10) COLLATE latin1_german1_ci DEFAULT NULL,
  `besitzer1840` text COLLATE latin1_german1_ci,
  `hausnr1810` varchar(10) COLLATE latin1_german1_ci DEFAULT NULL,
  `besitzer1810` text COLLATE latin1_german1_ci,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIME-
STAMP,
  PRIMARY KEY (`ID`),
  UNIQUE KEY `plannr` (`plannr`),
  KEY `hausnr1913` (`hausnr1913`),
  KEY `strassenrheute` (`strassenrheute`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci ROW_
FORMAT=DYNAMIC;

```

```

--
-- Tabellenstruktur für Tabelle `hochzeiten`
--

CREATE TABLE IF NOT EXISTS `hochzeiten` (
  `id` int(6) NOT NULL AUTO_INCREMENT,
  `mann` varchar(255) COLLATE latin1_german1_ci NOT NULL,
  `jahr` int(4) DEFAULT NULL,
  `frau` varchar(255) COLLATE latin1_german1_ci NOT NULL,
  `mnachname` varchar(100) COLLATE latin1_german1_ci NOT NULL,
  `mvorname` varchar(100) COLLATE latin1_german1_ci NOT NULL,
  `mvornamegruppe` varchar(100) COLLATE latin1_german1_ci DEFAULT NULL,
  `mberuf` varchar(100) COLLATE latin1_german1_ci DEFAULT NULL,
  `mgebort` varchar(100) COLLATE latin1_german1_ci DEFAULT NULL,
  `mgebjahr` int(4) DEFAULT NULL,
  `mwitwer` varchar(1) COLLATE latin1_german1_ci NOT NULL DEFAULT 'n',
  `fnachname` varchar(100) COLLATE latin1_german1_ci NOT NULL,
  `fvorname` varchar(100) COLLATE latin1_german1_ci NOT NULL,
  `fvornamegruppe` varchar(20) COLLATE latin1_german1_ci DEFAULT NULL,
  `fberuf` varchar(100) COLLATE latin1_german1_ci DEFAULT NULL,
  `fgebort` varchar(100) COLLATE latin1_german1_ci DEFAULT NULL,
  `fgebjahr` int(4) DEFAULT NULL,
  `fwitwe` varchar(1) COLLATE latin1_german1_ci NOT NULL DEFAULT 'n',
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`),
  KEY `mgebort` (`mgebort`),
  KEY `fgebort` (`fgebort`),
  KEY `mvorname` (`mvorname`),
  KEY `fvorname` (`fvorname`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci
COMMENT='Hochzeitsbuch Kriegshaber 1600-1975';

```

```

-----

```

```

--
-- Tabellenstruktur für Tabelle `katabk`
--

```

```

CREATE TABLE IF NOT EXISTS `katabk` (
  `id` int(10) NOT NULL AUTO_INCREMENT,
  `abk` varchar(10) COLLATE latin1_german1_ci NOT NULL,
  `name` varchar(255) COLLATE latin1_german1_ci DEFAULT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Kataster-Abkuerzungen';

```

```

-----

```



```
--
-- Tabellenstruktur für Tabelle `katabk`
--

CREATE TABLE IF NOT EXISTS `katabk` (
  `id` int(10) NOT NULL AUTO_INCREMENT,
  `abk` varchar(10) COLLATE latin1_german1_ci NOT NULL,
  `name` varchar(255) COLLATE latin1_german1_ci DEFAULT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Kataster-Abkuerzungen';
```

```
-- -----
--
-- Tabellenstruktur für Tabelle `katbandn`
--
```

```
CREATE TABLE IF NOT EXISTS `katbandn` (
  `id` int(10) NOT NULL AUTO_INCREMENT,
  `raort` varchar(10) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `typ` varchar(20) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `bandnr` varchar(40) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `ksv` varchar(20) CHARACTER SET latin1 COLLATE latin1_german1_ci DEFAULT NULL,
  `ksb` varchar(20) CHARACTER SET latin1 COLLATE latin1_german1_ci DEFAULT NULL,
  `hv` varchar(20) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `hb` varchar(20) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

```
-- -----
--
-- Tabellenstruktur für Tabelle `katkopf`
--
```

```
CREATE TABLE IF NOT EXISTS `katkopf` (
  `id` int(10) NOT NULL AUTO_INCREMENT,
  `raort` varchar(100) COLLATE latin1_german1_ci NOT NULL,
  `rentamt` varchar(100) COLLATE latin1_german1_ci NOT NULL,
  `steuergemeinde` varchar(100) COLLATE latin1_german1_ci NOT NULL,
  `steuerdistrikt` varchar(100) COLLATE latin1_german1_ci NOT NULL,
  `landamtsger` varchar(100) COLLATE latin1_german1_ci NOT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`),
  KEY `raortabk` (`raort`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Kataster-Details';
```

```

--
-- Tabellenstruktur für Tabelle `kattypn`
--
CREATE TABLE IF NOT EXISTS `kattypn` (
  `id` int(10) NOT NULL AUTO_INCREMENT,
  `typ` varchar(20) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `gruppe` int(10) NOT NULL DEFAULT '0',
  `abk` varchar(40) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `jahrv` int(4) DEFAULT NULL,
  `jahrb` int(4) DEFAULT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-----
--
-- Tabellenstruktur für Tabelle `kreisentfernung`
--
CREATE TABLE IF NOT EXISTS `kreisentfernung` (
  `id` int(10) NOT NULL AUTO_INCREMENT,
  `kreis` varchar(255) COLLATE latin1_german1_ci NOT NULL,
  `entfernung` int(7) DEFAULT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`),
  KEY `entfernung` (`entfernung`),
  KEY `kreis` (`kreis`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Kreis Entfernung';

-----
--
-- Tabellenstruktur für Tabelle `namenvergleich`
--
CREATE TABLE IF NOT EXISTS `namenvergleich` (
  `id` int(5) NOT NULL AUTO_INCREMENT,
  `nachname1857` varchar(100) CHARACTER SET latin1 COLLATE latin1_german1_ci DEFAULT NULL,
  `vorname1857` varchar(100) CHARACTER SET latin1 COLLATE latin1_german1_ci DEFAULT NULL,
  `plannr` varchar(100) CHARACTER SET latin1 COLLATE latin1_german1_ci DEFAULT NULL,
  `hausnr1913` varchar(20) CHARACTER SET latin1 COLLATE latin1_german1_ci DEFAULT NULL,
  `besitzer1892` text CHARACTER SET latin1 COLLATE latin1_german1_ci,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `besitzer1867` text CHARACTER SET latin1 COLLATE latin1_german1_ci,
  `besitzer1854` text CHARACTER SET latin1 COLLATE latin1_german1_ci,
  `besitzer1840` text CHARACTER SET latin1 COLLATE latin1_german1_ci,
  PRIMARY KEY (`id`),
  KEY `namen1857` (`nachname1857`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-----

```

```
--
-- Tabellenstruktur für Tabelle `navigation`
--
CREATE TABLE IF NOT EXISTS `navigation` (
  `id` int(10) UNSIGNED NOT NULL AUTO_INCREMENT,
  `gruppe` char(2) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL DEFAULT '00',
  `ugruppe` char(2) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `aktiv` varchar(1) NOT NULL DEFAULT 'j',
  `titel` varchar(255) CHARACTER SET utf8 DEFAULT NULL,
  `text` text CHARACTER SET latin1 COLLATE latin1_german1_ci,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

```
-----
--
-- Tabellenstruktur für Tabelle `ngbdetail`
--
CREATE TABLE IF NOT EXISTS `ngbdetail` (
  `ID` int(4) NOT NULL AUTO_INCREMENT,
  `stgem` varchar(10) COLLATE latin1_german1_ci DEFAULT 'askri',
  `katseite` varchar(40) COLLATE latin1_german1_ci DEFAULT 'verschiedene',
  `plannr` varchar(50) COLLATE latin1_german1_ci DEFAULT 'verschiedene',
  `hausnr` varchar(40) COLLATE latin1_german1_ci DEFAULT 'H#092',
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `schl` varchar(6) COLLATE latin1_german1_ci DEFAULT 'spital',
  `lfdnr` varchar(3) COLLATE latin1_german1_ci DEFAULT '000',
  `art` varchar(255) COLLATE latin1_german1_ci DEFAULT 'Total Summe',
  `fgfls` decimal(10,0) DEFAULT NULL,
  `fgkrs` decimal(5,0) DEFAULT NULL,
  `fghls` decimal(5,0) DEFAULT NULL,
  `akfld` decimal(10,0) DEFAULT NULL,
  `akkrd` decimal(5,0) DEFAULT NULL,
  `akhld` decimal(5,0) DEFAULT NULL,
  `akfls` decimal(10,0) DEFAULT NULL,
  `akkrs` decimal(5,0) DEFAULT NULL,
  `akhls` decimal(5,0) DEFAULT NULL,
  `zafld` decimal(10,0) DEFAULT NULL,
  `zakrd` decimal(5,0) DEFAULT NULL,
  `zahld` decimal(5,0) DEFAULT NULL,
  `zafls` decimal(10,0) DEFAULT NULL,
  `zakrs` decimal(5,0) DEFAULT NULL,
  `zahls` decimal(5,0) DEFAULT NULL,
  `bemerkung` text COLLATE latin1_german1_ci,
  `quelle` varchar(255) COLLATE latin1_german1_ci DEFAULT '2258 I',
  PRIMARY KEY (`ID`),
  UNIQUE KEY `indexortschl` (`stgem`,`schl`,`katseite`,`hausnr`,`lfdnr`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci ROW_FORMAT=DYNAMIC;
```

```

--
-- Tabellenstruktur für Tabelle `ngbdetart`
--

CREATE TABLE IF NOT EXISTS `ngbdetart` (
  `ID` int(4) NOT NULL AUTO_INCREMENT,
  `stgem` varchar(10) COLLATE latin1_german1_ci NOT NULL DEFAULT 'askri',
  `katseite` varchar(20) COLLATE latin1_german1_ci NOT NULL DEFAULT 'verschiedene',
  `plannr` varchar(100) COLLATE latin1_german1_ci DEFAULT 'verschiedene',
  `hausnr` varchar(20) COLLATE latin1_german1_ci DEFAULT 'H#015',
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `schl` varchar(6) COLLATE latin1_german1_ci DEFAULT 'koberh',
  `art` varchar(255) COLLATE latin1_german1_ci DEFAULT 'Kleinzehent: ',
  `fgfl` decimal(10,0) DEFAULT NULL,
  `fgkr` decimal(5,0) DEFAULT NULL,
  `fghl` decimal(5,0) DEFAULT NULL,
  `bemerkung` text COLLATE latin1_german1_ci,
  `quelle` varchar(255) COLLATE latin1_german1_ci NOT NULL DEFAULT '252 II',
  PRIMARY KEY (`ID`),
  KEY `hausnr` (`hausnr`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci ROW_FORMAT=DYNAMIC;

```

```

--
-- Tabellenstruktur für Tabelle `ngbkrigem`
--

CREATE TABLE IF NOT EXISTS `ngbkrigem` (
  `id` int(4) NOT NULL AUTO_INCREMENT,
  `schl` varchar(10) COLLATE latin1_german1_ci NOT NULL,
  `kreis` varchar(100) COLLATE latin1_german1_ci NOT NULL,
  `gemeinde` varchar(100) COLLATE latin1_german1_ci NOT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`),
  KEY `schl` (`schl`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Neues Grundbuch Ablösungskasse Abkürzung Kreis Gemeinde';

```

```

--
-- Tabellenstruktur für Tabelle `ngbschl`
--

CREATE TABLE IF NOT EXISTS `ngbschl` (
  `id` int(10) NOT NULL AUTO_INCREMENT,
  `schl` varchar(6) COLLATE latin1_german1_ci NOT NULL,
  `besitzer` varchar(255) COLLATE latin1_german1_ci NOT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`),
  UNIQUE KEY `schl` (`id`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Neues Grundbuch: Schlüssel der Lehensträger';

-----

```

```

--
-- Tabellenstruktur für Tabelle `ortkreis`
--

CREATE TABLE IF NOT EXISTS `ortkreis` (
  `ort` varchar(255) COLLATE latin1_german1_ci NOT NULL,
  `kreis` varchar(255) COLLATE latin1_german1_ci DEFAULT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`ort`),
  KEY `kreis` (`kreis`)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Ort Kreis';

-----

```

```

--
-- Tabellenstruktur für Tabelle `st`
--

CREATE TABLE IF NOT EXISTS `st` (
  `stadtbeznr` int(2) NOT NULL DEFAULT '0',
  `stname` text CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`stadtbeznr`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

-----

```

```

--
-- Tabellenstruktur für Tabelle `stadtaabilder`
--
CREATE TABLE IF NOT EXISTS `stadtaabilder` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `titel` varchar(255) COLLATE latin1_german1_ci NOT NULL,
  `strasse` varchar(100) COLLATE latin1_german1_ci DEFAULT "",
  `hausnr` varchar(100) COLLATE latin1_german1_ci DEFAULT "",
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `datierung` varchar(100) COLLATE latin1_german1_ci DEFAULT "",
  `positiv` varchar(100) COLLATE latin1_german1_ci DEFAULT "",
  `dia` varchar(100) COLLATE latin1_german1_ci DEFAULT "",
  `beschreibung` text COLLATE latin1_german1_ci,
  `text` text COLLATE latin1_german1_ci NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci
COMMENT='Dokumentation Katasterfunktionen';
-----

--
-- Tabellenstruktur für Tabelle `stadtaakriegshaber`
--
CREATE TABLE IF NOT EXISTS `stadtaakriegshaber` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `titelnummer` varchar(10) COLLATE latin1_german1_ci NOT NULL,
  `titel` varchar(100) COLLATE latin1_german1_ci NOT NULL,
  `nummer` varchar(100) COLLATE latin1_german1_ci NOT NULL,
  `bemerkung` text COLLATE latin1_german1_ci NOT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci
COMMENT='Dokumentation für verarbeitete Kirchenbücher' PACK_KEYS=0;
-----

--
-- Tabellenstruktur für Tabelle `statamtstrassen`
--
CREATE TABLE IF NOT EXISTS `statamtstrassen` (
  `id` int(4) NOT NULL AUTO_INCREMENT,
  `schluessel` int(10) NOT NULL,
  `strname` text COLLATE latin1_german1_ci NOT NULL,
  `stadtbeznr` int(2) NOT NULL DEFAULT '18',
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`schluessel`),
  UNIQUE KEY `id` (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Amt für
Statistik Straßen 18 mit Schlüssel';
-----

```

```
--
-- Tabellenstruktur für Tabelle `strassen`
--

CREATE TABLE IF NOT EXISTS `strassen` (
  `id` int(4) NOT NULL AUTO_INCREMENT,
  `schluessel` int(10) NOT NULL,
  `strassenr1` int(4) NOT NULL DEFAULT '0',
  `strassenr2` int(4) NOT NULL DEFAULT '0',
  `strassenr3` int(4) NOT NULL DEFAULT '0',
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`),
  KEY `istrnr1` (`strassenr1`),
  KEY `istrnr2` (`strassenr2`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Referenz der Straßennummern';
```

```
--
-- Tabellenstruktur für Tabelle `strassen1`
--
```

```
CREATE TABLE IF NOT EXISTS `strassen1` (
  `id` int(4) NOT NULL AUTO_INCREMENT,
  `schluessel` int(10) NOT NULL,
  `strnralt` int(4) NOT NULL DEFAULT '0',
  `strassenname` text CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `stadtbeznr` int(2) NOT NULL DEFAULT '18',
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`),
  KEY `strassenr` (`strnralt`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

```

--
-- Tabellenstruktur für Tabelle `strassen2`
--

CREATE TABLE IF NOT EXISTS `strassen2` (
  `id` int(4) NOT NULL AUTO_INCREMENT,
  `strnralt` int(4) NOT NULL DEFAULT '0',
  `schluessel` int(10) NOT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  `status1913` tinyint(1) NOT NULL DEFAULT '0',
  `status1938` tinyint(1) NOT NULL DEFAULT '0',
  `status2008` tinyint(1) NOT NULL DEFAULT '0',
  `bemerkung` text CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `foto1` text CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `foto2` text CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `foto3` text CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `verknuepfung1` int(5) NOT NULL DEFAULT '0',
  `verknuepfung2` int(5) DEFAULT '0',
  `falk2001` varchar(20) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `falk2008` varchar(20) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  PRIMARY KEY (`id`),
  KEY `schluessel` (`schluessel`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1;

```

```

--
-- Tabellenstruktur für Tabelle `strassenstadt`
--

```

```

CREATE TABLE IF NOT EXISTS `strassenstadt` (
  `id` int(4) NOT NULL AUTO_INCREMENT,
  `strassennr1` int(4) NOT NULL DEFAULT '0',
  `strschl` int(4) NOT NULL DEFAULT '0',
  `strassenname` varchar(50) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `stadtnr` int(2) NOT NULL DEFAULT '0',
  `stadtbezirk` varchar(50) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`),
  UNIQUE KEY `istrassennr1` (`strassennr1`),
  KEY `istadtbez` (`stadtnr`),
  KEY `istrassenname` (`strassenname`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Excel Tabelle des Vermessungsamtes';

```

```
--
-- Tabellenstruktur für Tabelle `taufen`
--

CREATE TABLE IF NOT EXISTS `taufen` (
  `id` int(4) NOT NULL AUTO_INCREMENT,
  `seite` varchar(10) NOT NULL,
  `jahr` int(4) NOT NULL DEFAULT '0',
  `anzahl` int(4) NOT NULL DEFAULT '0',
  `monate` varchar(100) CHARACTER SET utf8 NOT NULL DEFAULT '0',
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COMMENT='Taufen Summen pro Jahr';

-----

--
-- Tabellenstruktur für Tabelle `temp`
--

CREATE TABLE IF NOT EXISTS `temp` (
  `summeanzahl` bigint(21) NOT NULL DEFAULT '0',
  `jahr` int(4) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-----

--
-- Tabellenstruktur für Tabelle `tmp`
--

CREATE TABLE IF NOT EXISTS `tmp` (
  `plannr` varchar(100) CHARACTER SET latin1 COLLATE latin1_german1_ci DEFAULT NULL,
  `hausnr1913` varchar(20) CHARACTER SET latin1 COLLATE latin1_german1_ci DEFAULT NULL,
  `hausnr1918` varchar(100) CHARACTER SET latin1 COLLATE latin1_german1_ci DEFAULT NULL,
  `hausnrheute` varchar(100) CHARACTER SET latin1 COLLATE latin1_german1_ci DEFAULT NULL,
  `anzahl` bigint(21) NOT NULL DEFAULT '0'
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

-----
```

Das sind alle Tabellen der Datenbank kriegshaber

N.B. Es gibt noch viele andere Tabellen, die praktisch nur aus einem Feld text bestehen. Diese Tabellen wurden natürlich nicht alle ausgedruckt. Sie sind identisch mit der Tabelle buchsrbuch, die ganz am Anfang steht, also 3 Felder.

Beispiele für SELECT-Befehle:

Einfacher Select: eine Tabelle sortiert ausgeben:

```
SELECT mann, frau, jahr  
FROM hochzeiten  
ORDER BY frau, jahr
```

Select, um eine Tabelle mit nur einem Satz anzuzeigen:

```
SELECT text  
FROM dokukatasterdaten
```

Die Sätze einer Tabelle zählen und ausgeben nach Gruppen pro Jahr:

```
SELECT jahr, count(*) AS summejahr  
FROM hochzeiten  
GROUP BY jahr  
ORDER BY jahr
```

Die Straßennamen sortiert ausgeben:

```
SELECT strassenname, stadtbnr, stadtbezirk  
FROM strassenstadt  
ORDER BY strassenname
```

Alle Sätze der Tabelle Hochzeiten zählen und pro Beruf der Frau summiert ausgeben

```
SELECT fberuf, count(*) AS summeberuf  
FROM hochzeiten  
GROUP BY fberuf  
ORDER BY fberuf
```

Die Vorname der männlichen Hochzeiter ausgeben:

```
SELECT mann, mvorname  
FROM hochzeiten  
ORDER BY mvorname
```

Befehl für das Navigations-Programm:

```
SELECT gruppe , ugruppe , titel  
FROM navigation  
WHERE gruppe = '00'  
AND aktiv = 'j'  
ORDER BY ugruppe
```

Select um das Untermenü 30 aufzurufen:

```
SELECT gruppe , ugruppe , titel  
FROM navigation  
WHERE gruppe = '30'  
AND aktiv = 'j'  
GROUP BY ugruppe
```

Programm, um alle Häuser zu zeigen mit der verknüpften Tabelle statamtstrassen:

```
SELECT haeuser.hausnr1913, haeuser.plannr, haeuser.schluessel1913, haeuser.schluessel1918,
haeuser.strassennr, statamtstrassen.strname AS strassenname2012, haeuser.hausnrneu, haeuser.besitzer1913
FROM haeuser
LEFT JOIN statamtstrassen ON haeuser.strassennr = statamtstrassen.schluessel ORDER BY hausnr1913
```

Befehl, um alle Katasterbände anzuzeigen

```
SELECT DISTINCT katkopf.raort AS raort, katkopf.rentamt AS ra, katkopf.steuergemeinde AS stg, katkopf.
steuerdistrikt AS std, katkopf.landamtsger AS ger,
katabk.name, katabk.abk AS abk, kattyp.gruppe AS gruppe, katband.bandnr AS bnr, katband.ksv AS ksv,
katband.ksb AS ksb, katband.hv AS hv,
katband.hb AS hb, kattyp.jahrv AS jahrv, kattyp.jahrb AS jahrb
FROM katband LEFT JOIN katkopf ON katkopf.raort = katband.raort
LEFT JOIN katabk ON katband.abk = katabk.abk
LEFT JOIN kattyp ON katband.abk = kattyp.abk
WHERE katabk.abk IS NOT NULL AND katband.abk = kattyp.abk AND katband.raort = kattyp.raort
ORDER BY katkopf.rentamt, katkopf.steuergemeinde, katabk.abk, katband.ksv
```

Anzeigen von Personen mit diversen Daten (u.a. Alter), die Variablen \$fam, \$jahr1 und \$jahr2 werden vorher eingegeben

```
SELECT geburtdetail.fam AS fam, knachname, kvorname, kgebdatum, DATE_Format(kgebdatum, '%Y')
AS kgebjahr, quelle, ktoddatum,
(YEAR(ktoddatum) - YEAR(kgebdatum)) AS kalterj,
(MONTH(ktoddatum) - MONTH(kgebdatum)) AS kalterm,
(DAY(ktoddatum) - DAY(kgebdatum)) AS kaltert,
vnachname, vvorname, vgebjahr, DATE_Format(kgebdatum, '%Y') - vgebjahr AS valter,
mnachname, mvorname, mgebjahr, DATE_Format(kgebdatum, '%Y') - mgebjahr AS malter
FROM `geburtdetail`
WHERE ktoddatum IS NOT NULL
AND kgebdatum IS NOT NULL
AND (YEAR(ktoddatum) - YEAR(kgebdatum)) < 1
AND fam = '$fam'
AND DATE_Format(kgebdatum, '%Y') >= $jahr1
AND DATE_Format(kgebdatum, '%Y') <= $jahr2
OR ktoddatum IS NOT NULL
AND kgebdatum IS NOT NULL
AND (YEAR(ktoddatum) - YEAR(kgebdatum)) < 2
AND (MONTH(ktoddatum) - MONTH(kgebdatum)) < 0
AND fam = '$fam'
AND DATE_Format(kgebdatum, '%Y') >= $jahr1
AND DATE_Format(kgebdatum, '%Y') <= $jahr2
ORDER BY knachname, kvorname
```

Anzeigen von Personen, die nicht ein Jahr geworden sind

```
SELECT geburtendetail.fam AS fam, knachname, kvorname, kgebdatum, DATE_Format(kgebdatum, '%Y')
AS kgebjahr, quelle, ktoddatum,
(YEAR(ktoddatum) - YEAR(kgebdatum)) AS kalterj,
(MONTH(ktoddatum) - MONTH(kgebdatum)) AS kalterm,
(DAY(ktoddatum) - DAY(kgebdatum)) AS kaltert,
vnachname, vvorname, vgebjahr, DATE_Format(kgebdatum, '%Y') - vgebjahr AS valter,
mnachname, mvorname, mgebjahr, DATE_Format(kgebdatum, '%Y') - mgebjahr AS malter
FROM `geburtendetail`
WHERE ktoddatum IS NOT NULL
AND kgebdatum IS NOT NULL
AND (YEAR(ktoddatum) - YEAR(kgebdatum)) < 1
OR ktoddatum IS NOT NULL
AND kgebdatum IS NOT NULL
AND (YEAR(ktoddatum) - YEAR(kgebdatum)) < 2
AND (MONTH(ktoddatum) - MONTH(kgebdatum)) < 0
ORDER BY knachname, kvorname
```

Anzeigen von Personen, die leben und heute über 20 Jahre sind und auch diejenigen, die in einem Alter über 20 Jahre verstorben sind

```
SELECT geburtendetail.fam AS fam, knachname, kvorname, kgebdatum, DATE_Format(kgebdatum, '%Y')
AS kgebjahr, quelle, ktoddatum,
(YEAR(ktoddatum) - YEAR(kgebdatum)) AS kalterj,
(MONTH(ktoddatum) - MONTH(kgebdatum)) AS kalterm,
(DAY(ktoddatum) - DAY(kgebdatum)) AS kaltert,
vnachname, vvorname, vgebjahr, DATE_Format(kgebdatum, '%Y') - vgebjahr AS valter,
mnachname, mvorname, mgebjahr, DATE_Format(kgebdatum, '%Y') - mgebjahr AS malter
FROM `geburtendetail`
WHERE ktoddatum IS NULL
AND kgebdatum IS NOT NULL
AND (DATE_FORMAT(CURDATE(), '%Y') - DATE_FORMAT(kgebdatum, '%Y')) > 20
OR ktoddatum IS NOT NULL
AND kgebdatum IS NOT NULL
AND (YEAR(ktoddatum) - YEAR(kgebdatum)) > 20
ORDER BY knachname, kvorname
```

Doppelte Sätze rausfiltern:

```
DROP TABLE If EXISTS tmp;
CREATE TABLE tmp SELECT plannr, hausnr1913, hausnrneu1918, hausnrheute,
count(*) as anzahl
FROM haeuser
group by $feld;
SELECT plannr, hausnr1913, hausnrneu1918, hausnrheute, anzahl
FROM tmp
WHERE anzahl > 1;
```

Doppelte Sätze rausfiltern:

```
DROP TABLE If EXISTS tmp;
```

```
CREATE TABLE tmp SELECT plannr, hausnr1913, hausnrneu1918, hausnrheute,  
count(*) as anzahl
```

```
FROM haeuser
```

```
group by $feld;
```

```
SELECT plannr, hausnr1913, hausnrneu1918, hausnrheute, anzahl
```

```
FROM tmp
```

```
WHERE anzahl > 1;
```

Beispiel, wie ich untersuche, ob es doppelte Steuerzeichen (gruppe und ugruppe) in der Datei navigation gibt:

```
DROP TABLE If EXISTS tmp
```

```
CREATE TABLE tmp SELECT gruppe, ugruppe, count(*) as anzahl
```

```
FROM navigation
```

```
SELECT gruppe, ugruppe, anzahl
```

```
FROM tmp
```

```
WHERE anzahl > 1
```

```
?>
```

-- Anhang zu Buch Die Datenbank MySQL als Hilfsmittel für die -- Darstellung geschichtlicher Themen:
Datei FORMAT.CSS

```
/* veränderte Stildatei, Stand 24. Juli 2004, Original von Adresse http://www.knowware.de/kwnew/css/  
knowware.css*/
```

```
/* Schriftart im gesamten Dokument */
```

```
body {  
font-family: Arial, Verdana, Helvetica, sans-serif;  
margin-top: 0px;  
background-color: #98FB98; // #FFFF00 gelb; #98FB98 hellgrün; #40E0D0 türkis; #FFE4C4 bisque;  
}
```

```
.tbl_gesamt {  
border-style: solid;  
border-width: 1px;  
border-color: black;  
background-color: #FFE4C4; // #FFFF00 gelb; #98FB98 hellgrün; #40E0D0 türkis; #FFE4C4 bisque;
```

```
}
```

```
/* Überschrift 1 background color mitternachtblau */
```

```
h1 {  
font-family: Verdana, Arial, Helvetica, sans-serif;  
font-size: 11pt;  
color: white;  
background-color: #CD853F; // mittelbraun  
margin-top: 0px;  
padding-top: 4px;  
margin-bottom: 6px;  
text-transform: uppercase;  
padding-bottom: 4px;  
margin-left: 4px;  
margin-right: 4px;  
text-align: center;  
margin-bottom: 8px;  
}
```

```
/* Überschrift 2 und 3 blau färben */
```

```
h2 {  
color: #0070B0;  
font-size: 12pt;  
margin-bottom: -2px;  
}
```

```
/* Überschrift 3 blau färben */
```

```
h3 {  
font-size: 12pt;  
margin-bottom: 2px;  
color: #0070B0;  
}
```

```
h4 {
font-size: 11pt;
margin-top: 2px;
margin-bottom: 2px;
color: #4E4E4E;
}
```

```
h5 {
font-size: 10pt;
margin-top: 4px;
margin-bottom: 2px;
margin-left: 4px;
}
```

```
/* DIV-Container für Welcome-Text */
div.welcome, div.welcome p {
font-family: Verdana, Arial, Helvetica, sans-serif;
font-size: 10pt;
font-weight: bold;
margin-top: 12px;
margin-left: 6px;
margin-right: 6px;
margin-bottom: 20px;
line-height: 1.2;
}
```

```
div.welcome ul {
list-style-image: url(reddot.gif);
}
```

```
div.rightbody {
margin-left: 3px;
margin-right: 5px;
}
```

```
div.right {
font-family: Verdana, Arial, Helvetica, sans-serif;
font-size: 10pt;
/* font-weight: bold; */
color: #80FF80;
margin-top: 0px;
margin-left: 2px;
margin-right: 2px;
margin-bottom: 10px;
}
```

```
div.right a, div.right a:visited {  
color: red;  
}
```

```
div.right a:hover {  
color: #FF0000;  
}
```

```
/* DIV-Container für Hefte */  
div.prod {  
margin-top: 0px;  
margin-left: 7px;  
margin-right: 7px;  
}
```

```
.redbar {  
font-family: Verdana, Arial, Helvetica, sans-serif;  
font-size: 11pt;  
font-weight: bold;  
color: white;  
background-color: #DF2016; // intensiv rot  
margin-bottom: 9px;  
height: 25px;  
text-transform: uppercase;  
padding-top: 4px;  
padding-bottom: 4px;  
text-align: center;  
border-bottom-style: solid;  
border-bottom-width: 5px;  
border-bottom-color: white;  
}
```

```
.farbmini {  
background-color: #98FB98; // hellgrün  
font-size: 8pt;  
padding: 3px;  
}
```

```
/* Minischrift */  
.mini {  
font-family: Verdana, Arial, Helvetica, sans-serif;  
font-size: 8pt;  
color: black;  
}
```

```
/* Minischrift, großer linker Rand */  
.mini2 {  
font-family: Verdana, Arial, Helvetica, sans-serif;  
font-size: 8pt;  
margin-left: 5px;  
}
```



```
.title {  
font-family: Verdana, Arial, Helvetica, sans-serif;  
font-size: 15pt;  
font-weight: bold;  
color: white;  
}
```

```
div.lead {  
font-family: Verdana, Arial, Helvetica, sans-serif;  
font-size: 10pt;  
margin-top: 5px;  
margin-bottom: 7px;  
}
```

```
div.pdfink {  
font-family: Verdana, Arial, Helvetica, sans-serif;  
font-size: 9pt;  
text-align: center;  
padding: 5px;  
background-color: #FFF2F2; // ganz hell rosa  
}
```

```
/* Kategorien */  
div.subcat {  
font-family: Verdana, Arial, Helvetica, sans-serif;  
font-size: 9pt;  
text-align: left;  
color: silver;  
padding-left: 8px;  
}
```

```
a.subcat:link {  
color: silver;  
text-decoration: none;  
}
```

```
a.subcat:visited, a.subcat:active {  
font-family: Verdana, Arial, Helvetica, sans-serif;  
text-decoration: none;  
text-align: center;  
color: silver;  
}
```

```
a.subcat:hover {  
font-family: Verdana, Arial, Helvetica, sans-serif;  
text-decoration: none;  
color: white;  
}
```

```
div.subcactive {
font-family: Verdana, Arial, Helvetica, sans-serif;
font-size: 9pt;
text-decoration: none;
text-align: left;
color: white;
background-color: #98FB98; // hellgrün
padding: 1px;
padding-left: 8px;
}
```

```
a.cat:link, a.cat:visited, a.cat:active {
text-decoration: none;
color: white;
}
```

```
a.cat:hover {
text-decoration : underline;
color: white;
}
```

```
div.cactive {
font-family: Verdana, Arial, Helvetica, sans-serif;
color: black;
background-color: #98FB98; // hellgrün
font-size: 11px;
font-weight: bold;
text-transform: uppercase;
margin-top: 12px;
margin-bottom: 5px;
text-decoration: none;
text-align: left;
padding: 1px;
padding-left: 4px;
}
```

```
div.cat {
font-family: Verdana, Arial, Helvetica, sans-serif;
color: white;
font-size: 11px;
font-weight: bold;
text-decoration: none;
text-transform: uppercase;
margin-top: 12px;
margin-bottom: 5px;
text-align: left;
padding-left: 4px;
}
/* Kategorien Ende */
```

```
/* Absätze, Tabellen, Formulare in 10,5 Punkt */
```

```
p, div, table, li, form {  
font-size: 10pt;  
}
```

```
/* fette Passagen rot einfärben */
```

```
b {  
color: black;  
}
```

```
strong {  
color: black;  
}
```

```
span.product {  
font-weight: bold;  
}
```

```
/* kursive Passagen blau färben */
```

```
i {  
font-style: normal;  
}
```

```
/* Link-Stile für Hover-Links */
```

```
/* Pseudo-Klassen für dynamische Links */
```

```
a:visited {  
color: #10178C;  
text-decoration: underline;  
}
```

```
a:link {  
text-decoration : underline;  
color: #10178C;  
}
```

```
a:hover {  
text-decoration : underline;  
color : #10178C;  
}
```

```
a:active {  
text-decoration : underline;  
color : #10178C;  
}
```

```
.navi {  
color: white;  
}
```

```
a.navi:visited {
color: white;
font-weight: bold;
text-decoration: none;
}
```

```
a.navi:link {
color: white;
font-weight: bold;
text-decoration: none;
}
```

```
a.navi:hover {
color: white;
font-weight: bold;
text-decoration: underline;
}
```

```
/* Eingabefeld Notizzettel */
input.acht {
font-family : Verdana, Arial, Helvetica, sans-serif;
font-size : 8pt;
margin-left: -1px;
background-color: white;
border-style: solid;
border-width: 1px;
border-color: gray;
}
```

```
/* freie Vorlage, steuert Warenkorb und Kleingedrucktes */
.acht {
font-family : Verdana, Arial, Helvetica, sans-serif;
font-size : 9pt;
margin-left : 2px;
margin-bottom: 2px;
}
```

```
/* Bestellschein-Formular */
form.acht {
font-family : Verdana, Arial, Helvetica, sans-serif;
font-size : 10pt;
margin-left : 4px;
margin-top: 5px;
margin-bottom: 5px;
}
```

```
/* normaler Rahmen mit roter Umrandung, 8pt */
```

```
.rahmen {  
font-family : Verdana, Arial, Helvetica, sans-serif;  
font-size : 8pt;  
background-color : white;  
border-style : solid;  
border-width : 1px;  
border-color : silver;  
padding : 3px;  
margin-left : 3px;  
}
```

```
.tipp {  
font-family : Verdana, Arial, Helvetica, sans-serif;  
font-size : 9.5pt;  
border-style : dotted;  
border-width : 1px;  
border-color : blue;  
padding : 2px;  
}
```

```
div.catbar {  
color: #ffffff;  
background-color: #0070B0;  
}
```

```
td.linkbar {  
font-family: Verdana, Arial, Helvetica, sans-serif;  
color: #ffffff;  
background-color: #0070B0;  
font-weight: bold;  
text-align: center;  
text-transform: uppercase;  
height: 25px;  
}
```

```
td.linkbaractive {  
font-family: Verdana, Arial, Helvetica, sans-serif;  
color: #ffffff;  
background-color: #10178C;  
font-weight: bold;  
text-align: center;  
text-transform: uppercase;  
}
```

```
td.linkbar2 {
color: white;
background-color: #B2B2B2;
font-weight: bold;
text-align: center;
}
```

```
td.linkbaractive2 {
color: white;
background-color: #0033cc;
font-weight: bold;
text-align: center;
}
```

```
table.ordertable {
margin-left: 4px;
margin-right: 4px;
}
```

```
ul {
list-style-image: url(reddot.gif);
}
```

```
div.prod ul {
list-style-image: url(reddot.gif);
}
```

```
table.right {
}
```

```
table.adresse {
background-color: white;
}
```

```
td.dark {
background-color: silver;
}
```

```
td.light {
background-color: #fcffe6;
}
```

```
/* Inhalt des Heftes */
```

```
div.content {
font-family: Verdana, Arial, Helvetica, sans-serif;
font-size: 9pt;
background-color: #98FB98; // hellgrün
margin-top: 7px;
padding: 5px;
line-height: 1.3;
}
```

```
div.content p {
font-size: 9pt;
margin-top: 2px;
margin-bottom: 5px;
}
```

```
div.content h5 {
font-size: 9pt;
margin-left: 0px;
margin-top: 10px;
margin-bottom: 2px;
}
```

```
div.content ul {
font-size: 9pt;
list-style-image: (reddot.gif);
margin-top: 2px;
margin-bottom: 2px;
}
```

```
div.content li {
font-size: 9pt;
}
```

```
/* Minitabelle für Heftinfos */
```

```
table.heftinfos {
background-color: red;
}
```

```
table.heftinfos td {
background-color: white;
}
```

```
td.bookpointer {
padding: 3px;
border-style: dotted;
border-width: 3px;
border-color: silver;
background-color: white;
}
```

```
td.bookpointer ul {
margin-top: 2px;
margin-bottom: 3px;
margin-left: 15px;
padding-left: 0px;
}
```

Ende der Format-Datei

Datei gestaltung.txt

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("FORMATE.CSS") screen;
  </style>
  <meta name="author" content="Heinz Wember Augsburg, php7, 27-Feb-2019">
  <title>Gestaltung</title>
</head>
<body>
<left>
<br>
<center><strong><big><big>Textgestaltung mit html</big></big></center></strong><br>
<br>
<table style="width: 100%; text-align: left; margin-left: left; margin-right: auto;" border="0" cellpadding="2" cellspacing="2">
  <tbody>
<?php
  include "../db/zugriff-dbkriegshaber.inc.php";
  $abfrage = "SELECT text FROM `gestaltung`";
  $ergebnis = mysqli_query($db, $abfrage);
?>

<?php
  while($row = $ergebnis->fetch_assoc())
  {
    echo "<tr><td> $row[text]</td></tr>";
  }
  mysqli_close($db);
?>

  </tbody>
</table>
<br>
</body>
</html>
```


Programm hauseval1918.php

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("../FORMATE.CSS") screen;
    body { font-family:Arial,sans-serif; }
    body { background-color: #98FB98; } // #FFFF00 gelb; #98FB98 hellgrün; #40E0D0 türkis;
    a:link { color:#000000; text-decoration:underline;}
    a:visited { color:#FF0000; text-decoration:underline; }
    a:focus { color:#00AA00; background-color:#80FF80;}
    a:hover { color:#EE0000; text-decoration:none; background-color:#FFFF99; }
    a:active { color:#0000EE; background-color:#FFFF99;}
  </style>
  <meta name="author" content="Heinz Wember Augsburg, 02-Mai-2012">
  <title>Stra&szlig;en in Kriegshaber</title>
</head>
<body>
<br>
<h1>Straßen von Kriegshaber<BR>Fehlerreport: Alle Straßennummern in Tabelle haeuser (Feld schlues-
sell1918), die in Tabelle statamtstrassen nicht vorkommen </h1>
<br>
<?php
define("UNLOCK",1);
include "../db/zugriff-dbkriegshaber.inc.php";

mysql_select_db(MYSQL_DATENBANK) or die(mysql_error());
$abfrage = "SELECT haeuser.ID, haeuser.plannr, haeuser.hausnr1913, haeuser.strassenr1918, haeuser.
besitzer1913
FROM haeuser
LEFT JOIN statamtstrassen
ON haeuser.strassenr1918 = statamtstrassen.schluesel
WHERE statamtstrassen.schluesel is NULL
ORDER BY plannr";
$ergebnis = mysql_query($abfrage);
while($row = mysql_fetch_object($ergebnis))
{
echo "<BR>\nID= $row->ID";
echo "; plannr= $row->plannr";
echo "; hausnr1913= $row->hausnr1913";
echo "; strassenr1918= $row->schluesel1918";
echo "; besitzer1913 = $row->besitzer1913";
}
?>
<hr>
<center>
<a target="_blank" href="../index-oeff.html">Übersicht Daten unter Passwortschutz</a>
</center>
<hr>
<small>Änderungsstand: 27-Nov-2018<br>
<a href="../Impressum.html">Heinz Wember</a></body></html>
```

Programm index.htm

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/
DTD/xhtml1-transitional.dtd">
<html>
<head>
<style type="text/css">
  @import url("FORMATE.CSS") screen;
</style>
<title>Straßen, Häuser und Kataster von Kriegshaber</title>
<meta name="author" content="Heinz Wember Augsburg">
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
</head>

<frameset rows="130,*">
<!-- Dadurch ergeben sich zwei Frame-Fenster, deren Inhalt hier bestimmt wird.
  Das erste Frame-Fenster wird normal definiert, anstelle des zweiten
  folgt jedoch das folgende untergeordnete Frameset: -->
<frame name="top" src="top.php" scrolling="no" frameborder="no">
<frameset cols="15%,85%"rows=1200 >
<frame name="navigation" src="navigation.php" scrolling="auto" frameborder="no" >
<frame name="content" src="strassennamen.php " scrolling="auto" frameborder="no">
</frameset>
</frameset>

<body>
Datei index.html

</body>
</noframes>
</body>
</html>
```

Programm katband.php:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("FORMATE.CSS") screen;
  </style>

  <meta name="author" content="Heinz Wember Augsburg">
  <title>Staatsarchiv Augsburg: Katasterbände, Sort: Steuergemeinde, Gruppe, Typ</title>
</head>
<body>
<?php
  define("UNLOCK",1);
  include "../db/zugriff-dbkriegshaber.inc.php";
  $abfrage1 = "SELECT katbandn.id AS id, katkopf.raort AS raort, katkopf.rentamt AS ra,
    katkopf.steuergemeinde AS stg, katkopf.steuerdistrikt AS std,
    katkopf.landamtsger AS ger,
    katabk.name, katabk.abk AS abk,
    katbandn.typ AS typ, kattygn.gruppe AS gruppe,
    katbandn.bandnr AS bnr, katbandn.ksv AS ksv, katbandn.ksb AS ksb,
    katbandn.hv AS hv, katbandn.hb AS hb, kattygn.jahrv AS jahrv, kattygn.jahrb AS jahrb
  FROM katbandn
  LEFT JOIN katkopf ON katkopf.raort = katbandn.raort
  LEFT JOIN kattygn ON katbandn.typ = kattygn.typ
  LEFT JOIN katabk ON kattygn.abk = katabk.abk
  ORDER BY katkopf.steuergemeinde, kattygn.gruppe, kattygn.typ, katbandn.ksv, katbandn.band-
nr";
  $ergebnis1 = mysqli_query($db, $abfrage1);
?>

<center><br>
<h1>Staatsarchiv Augsburg: Quellenbände, Sort: Steuergemeinde, Gruppe, Typ</h1>
<h4>Kataster: Gruppe 1 bis 13; Gruppe = 0 andere Quellen (Grundbuch, Neues Grundbuch, Gefälle)</h4>
</center>
<left>
<br>
<table style="width: 100%; text-align: left; margin-left: left; margin-right: auto;" border="1" cellpadding="2" cellspacing="2">
  <tbody>
    <tr>
      <td style="width: 10%;>ID</td>
      <td style="width: 10%;>Rentamt</td>
      <td style="width: 10%;>Steuergemeinde</td>
      <td style="width: 10%;>Steuerdistrikt</td>
      <td style="width: 10%;>Land- Amtsgericht</td>
      <td style="width: 3%;>Typ</td>
      <td style="width: 3%;>Gruppe</td>
      <td style="width: 4%;>Abk</td>
      <td style="width: 25%;>Langtext zur Abk</td>
      <td style="width: 3%;>Jahr von</td>
      <td style="width: 3%;>Jahr bis</td>
```

```

    <td style="width: 5%;">Band Nr</td>
    <td style="width: 3%;">K S von</td>
    <td style="width: 3%;">K S bis</td>
    <td style="width: 3%;">H Nr von</td>
    <td style="width: 3%;">H Nr bis</td>
</tr>

```

```

<?php
while($row = $ergebnis1->fetch_assoc())
{
    echo "<td style=width: 3%;>$row[id]</td>";
    echo "<td style=width: 6%;>$row[ra]</td>";
    echo "<td style=width: 6%;>$row[stg]</td>";
    echo "<td style=width: 6%;>$row[std]</td>";
    echo "<td style=width: 6%;>$row[ger]</td>";
    echo "<td style=width: 6%;>$row[typ]</td>";
    echo "<td style=width: 3%;>$row[gruppe]</td>";
    echo "<td style=width: 4%;>$row[abk]</td>";
    echo "<td style=width: 25%;>$row[name]</td>";
    echo "<td style=width: 4%;>$row[jahr]v</td>";
    echo "<td style=width: 4%;>$row[jahr]b</td>";
    echo "<td style=width: 5%;>$row[bnr]</td>";
    echo "<td style=width: 3%;>$row[ksv]</td>";
    echo "<td style=width: 3%;>$row[ksb]</td>";
    echo "<td style=width: 3%;>$row[hv]</td>";
    echo "<td style=width: 3%;>$row[hb]</td></tr>";
}
mysqli_close($db);
?>

```

```

</tbody>
</table>

```

```

<hr>
<center>
<a target="_blank" href="index.html">Übersicht</a>
</center>
<hr>

```

```

<small>Änderungsstand Programm: 27-Feb-2019<br>
<a target="_blank" href=" ../Impressum.html">Heinz Wember</a></small>
</body>
</html>

```

Programm mrimkrse001bis020.php

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("../FORMATE.CSS") screen;
  </style>
  <meta name="author" content="Heinz Wember Augsburg, php7, 25-Feb-2019">
  <title>Summe der Entfernung vom Geburtsort der Männer - 001 bis 020 km</title>
</head>
<body>
<br><center>
<h1>Summe der Entfernung vom Geburtsort der Männer - 001 bis 020 km</h1>
</center><left>
<br>
<?php
  define("UNLOCK",1);
  include "../db/zugriff-dbkriegshaber.inc.php";
  $abfrage1 = "SELECT ortkreis.kreis AS kreis1, kreisentfernung.entfernung AS entf1, count(*) AS maenne-
rimkreis, SUM(kreisentfernung.entfernung) AS maennerimkreissum
    FROM hochzeiten
    LEFT JOIN ortkreis ON hochzeiten.mgebort = ortkreis.ort
    LEFT JOIN kreisentfernung ON ortkreis.kreis = kreisentfernung.kreis
    WHERE kreisentfernung.entfernung > 0
    AND kreisentfernung.entfernung < 21
    GROUP BY kreis1";
  $ergebnis1 = mysqli_query($db, $abfrage1);
  $abfrage2 = "SELECT count(*) AS maennerimkreis2, SUM(kreisentfernung.entfernung) AS maennerim-
kreissum2
    FROM hochzeiten
    LEFT JOIN ortkreis ON hochzeiten.mgebort = ortkreis.ort
    LEFT JOIN kreisentfernung ON ortkreis.kreis = kreisentfernung.kreis
    WHERE kreisentfernung.entfernung > 0
    AND kreisentfernung.entfernung < 21";
  $ergebnis2 = mysqli_query($db, $abfrage2);
?>

<table style="width: 50%; text-align: left; margin-left: left; margin-right: auto;" border="1" cellpadding="2"
cellspacing="2">
  <caption></caption><tbody>
  <tr>
    <td style="width: 20%;">Kreis</td>
    <td style="width: 10%;">Einzelentfernung</td>
    <td style="width: 10%;">Anzahl Männer</td>
    <td style="width: 10%;">Männer mal Entfernung</td>
  </tr>
```

```

<?php
while($row = $ergebnis1->fetch_assoc())
{
    echo "<tr><td style=width: 20%;>$row[kreis1]</td>";
    echo "<td style=width: 10%;>$row[entf1]</td>";
    echo "<td style=width: 10%;>$row[maennerimkreis]</td>";
    echo "<td style=width: 10%;>$row[maennerimkreisum]</td></tr>";
}
?>

```

```

<table style="width: 50%; text-align: left; margin-left: left; margin-right: auto;" border="1" cellpadding="2" cellspacing="2">

```

```

<caption></caption><tbody>
<tr>
<td style="width: 20%;>Summe</td>
<td style="width: 10%;> </td>
<td style="width: 10%;>Anzahl Männer</td>
<td style="width: 10%;>Männer mal Entfernung</td>
</tr>

```

```

<?php
while($row = $ergebnis2->fetch_assoc())
{
    echo "<td style=width: 20%;>Summe </td>";
    echo "<td style=width: 10%;>Entfernung 1 bis 20 km </td>";
    echo "<td style=width: 10%;>$row[maennerimkreis2]</td>";
    echo "<td style=width: 10%;>$row[maennerimkreisum2]</td>";
}
mysqli_close($db);
?>

```

```

</tbody>
</table>

```

```

<hr>
<center>
<a target="_blank" href="../index-kb.html">Übersicht Kirchenbücher</a>
</center>
<hr>
<small>Änderungsstand: 14-Mai-2019<br>
<a href="../Impressum.html">Heinz Wember</a>
</small>
</body>
</html>

```

Programm navigation.php:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("FORMATE.CSS") screen;
  </style>

  <meta name="author" content="Heinz Wember Augsburg, php7, 16-Feb-20190">

</head>

<left>
<small>
<table width=15%; text-align: left; margin-left: auto; margin-right: auto;" border="1" cellpadding="2" cell-
spacing="2">
  <tbody>
<?php
  include "../db/zugriff-dbkriegshaber.inc.php";
  $abfrage = "SELECT gruppe , ugruppe , titel
             FROM navigation
             WHERE gruppe = '00' AND aktiv = 'j'
             ORDER BY ugruppe";
  $ergebnis = mysqli_query($db, $abfrage);
  while($row = $ergebnis->fetch_assoc())
  {
    echo "<tr> $row[titel] </tr> <br>";
    $gruppe = $row[gruppe];
  }
  mysqli_close($db);

?>
  </tbody>
</table>
</small>
</body>
</html>
```

Programm ngbdetart.php:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("FORMATE.CSS") screen;
  </style>
  <meta name="author" content="Heinz Wember Augsburg, php7, 28-Feb-2019">
  <title>Ablösung der Grundlasten in Kriegshaber: Gefälle-Arten</title>
</head>
<body>
<?php
  define("UNLOCK",1);
  include "../db/zugriff-dbkriegshaber.inc.php";
  $abfrage = "SELECT kreis, gemeinde, katseite, plannr, hausnr, ngbdetart.schl, art,
    fgfl, fgkr, fgfl,
    bemerkung,
    ngbschl.besitzer AS besitzer
  FROM ngbdetart
  LEFT JOIN ngbschl ON ngbschl.schl = ngbdetart.schl
  LEFT JOIN ngbkrigem ON ngbkrigem.schl = ngbdetart.stgem
  ORDER BY kreis, gemeinde, hausnr";
  $ergebnis = mysqli_query($db, $abfrage);
?>

<center><br>
<h1>Ablösung der Grundlasten in Kriegshaber: Gefälle-Arten</h1>
<h3> FG= fixierte Größe in Geld, AK=Ablösungskapital zum 18fachen Betrag, ZA=4%er Zins dieses Kapitals</h3>
<h3>fl= Gulden, kr=Kreuzer, hl=Heller; </h3>
</center>
<left>

<br>
<table style="width: 100%; text-align: left; margin-left: left; margin-right: auto;" border="1" cellpadding="2" cellspacing="2">
  <tbody>
    <tr>
      <td style="width: 05%;">Kreis</td>
      <td style="width: 05%;">Gemeinde</td>
      <td style="width: 08%;">Kat-Seite</td>
      <td style="width: 05%;">FlurNr</td>
      <td style="width: 08%;">HausNr</td>
      <td style="width: 02%;">Schl</td>
      <td style="width: 20%;">Art</td>
      <td style="width: 02%;">FGfl</td>
      <td style="width: 02%;">FGkr</td>
      <td style="width: 02%;">FGhl</td>
      <td style="width: 20%;">Bemerkung</td>
      <td style="width: 20%;">Lehensherr</td>
    </tr>
  </tbody>
</table>
```



```

<?php
while($row = $ergebnis->fetch_assoc())
{
    echo "<tr><td style=width: 05%;>$row[kreis]</td>";
    echo "<td style=width; 05%;>$row[gemeinde]</td>";
    echo "<td style=width; 08%;>$row[katseite]</td>";
    echo "<td style=width; 05%;>$row[plannr]</td>";
    echo "<td style=width; 08%;>$row[hausnr]</td>";
    echo "<td style=width: 02%;>$row[schl]</td>";
    echo "<td style=width: 20%;>$row[art]</td>";
    echo "<td style=width: 02%;>$row[fgfl]</td>";
    echo "<td style=width: 02%;>$row[fgkr]</td>";
    echo "<td style=width: 02%;>$row[fghl]</td>";
    echo "<td style=width: 20%;>$row[bemerkung]</td>";
    echo "<td style=width: 20%;>$row[besitzer]</td></tr>";
}
mysqli_close($db);
?>

</tbody>
</table>

<hr>
<center>
<a target="_blank" href="index-ngb.html">Übersicht Kriegshaber</a>
</center>
</body>
</html>

```

Programm progdatum2.php:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("FORMATE.CSS") screen;
    body { font-family:Arial,sans-serif; }
    a:link { color:#000000; text-decoration:underline;}
    a:visited { color:#FF0000; text-decoration:underline; }
    a:focus { color:#00AA00; background-color:#80FF80;}
    a:hover { color:#EE0000; text-decoration:none; background-color:#FFFF99; }
    a:active { color:#0000EE; background-color:#FFFF99;}
  </style>
  <title>Kirchenchor Kriegshaber Dreifaltigkeit</title>
  <meta name="author" content="Heinz Wember Augsburg, php7, 28-Feb-2019">
</head>
<body>
<left>
<br>
<center>
<big><big><strong>Kirchenchor Kriegshaber Dreifaltigkeit</big></big></strong>
</center><br>
<br>
<table style="width: 100%px; text-align: left; margin-left: left; margin-right: auto;" border="1" cellpadding="2" cellspacing="2">
  <tbody>
<?php
  include "../db/zugriff-dbchorkriegshaber.inc.php";
  $abfrage = "SELECT
    id , datum,
    zeit, kirche, programm
    FROM `termine`
    WHERE datum >= CURDATE()
    ORDER BY `datum`";
  $ergebnis = mysqli_query($db, $abfrage);
  while($row = $ergebnis->fetch_assoc())
  {
    echo "<tr> ";
    echo "<td style=width: 10%;> $row[datum]</td>";
    echo "<td style=width: 10%;> $row[zeit]</td>";
    echo "<td style=width: 10%;> $row[kirche]</td>";
    echo "<td style=width: 10%;> $row[programm]</td></tr>";
  }
  mysqli_close($db);
?>
</tbody></table>
<hr><center>
<a target="_blank" href="index.html">Übersicht</a>
</center>
<hr>
<small>Änderungsstand: 28-Feb-2019<br></small>
</body></html>
```

Programm suchnamen.php:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("FORMATE.CSS") screen;
    body { font-family:Arial,sans-serif; }
    body { background-color: #98FB98; } // #FFFF00 gelb; #98FB98 hellgrün; #40E0D0 türkis;
    a:link { color:#000000; text-decoration:underline;}
    a:visited { color:#FF0000; text-decoration:underline; }
    a:focus { color:#00AA00; background-color:#80FF80;}
    a:hover { color:#EE0000; text-decoration:none; background-color:#FFFF99; }
    a:active { color:#0000EE; background-color:#FFFF99;}
  </style>
  <meta name="author" content="Heinz Wember Augsburg, php7, 28-Feb-2019">
  <title>Suchen nach Begriffen (Namen, Berufe etc) in den Feldern für Eigentümer, sortiert Hausnummer
1913</title>
</head>
<body>
<br><center>
<h1> Suchen nach Begriffen (Namen, Berufe etc) in den Feldern für Eigentümer, sortiert Hausnummer
1913</h1><br>
</center><left>
<br>
<?php
  define("UNLOCK",1);
  include "../db/zugriff-dbkriegshaber.inc.php";
  if (!$_POST['gesendet'] && !$_POST['name'])
  {

?>

  <form method="POST" action="<?php echo $PHP_SELF ?>">
    <p>Name z.B. %Steppich% oder %Jakob Beisele% oder %Schuhmachermeister% oder %073a%:
    <input type="text" name="name" size="30">
    </p>
    <input type="submit" name="Submit" value="Submit">
    <input type="hidden" name="gesendet" value="1">
  </form>

<?php
  }
else {
  $name=$_POST[name];
  $abfrage1 = "SELECT crossreference.hausnr1913, crossreference.plannr, crossreference.schluesse11913,
crossreference.schluesse11918, statamtstrassen.strname AS strassenname1918,
          statamtstrassen.schluesse1, crossreference.hausnrneu, crossreference.besitzer1913, crossrefe-
rence.hausnrumschr, crossreference.besitzerumschr,
          crossreference.hausnr1892, crossreference.besitzer1892, crossreference.hausnr1867, crossrefe-
rence.besitzer1867, crossreference.hausnr1840,
          crossreference.besitzer1840,
          crossreference.hausnr1810,
```

```

crossreference.besitzer1810
    FROM crossreference
    LEFT JOIN statamtstrassen ON crossreference.schluessel1918 = statamtstrassen.schluessel
    Where crossreference.besitzer1913 LIKE '$name'
    OR crossreference.besitzerumschr LIKE '$name'
    OR crossreference.besitzer1892 LIKE '$name'
    OR crossreference.besitzer1867 LIKE '$name'
    OR crossreference.besitzer1854 LIKE '$name'
    OR crossreference.besitzer1840 LIKE '$name'
    OR crossreference.plannr LIKE '$name'
    ORDER BY hausnr1913";
echo "Auswahl für $name";
$ergebnis1 = mysqli_query($db, $abfrage1);
?>
<table style="width: 85%; text-align: left; margin-left: left; margin-right: auto;" border="1" cellpadding="2"
cellspacing="2">
<caption></caption><tbody> <tr>
    <td style="width: 3%;">HausNr<br> 1913</td>
    <td style="width: 3%;">Plannr</td>
    <td style="width: 3%;">Schl 1913</td>
    <td style="width: 3%;">Schl 1918</td>
    <td style="width: 10%;">StrassenN1918 Hausnr neu</td>
    <td style="width: 20%;">Eigentümer/Bewohner 1913/1917</td>
    <td style="width: 20%;">HNr<br>Umschr</td>
    <td style="width: 20%;">Eigentümer Umschr</td>
    <td style="width: 20%;">HNr<br>1892</td>
    <td style="width: 20%;">Eigentümer 1892</td>
    <td style="width: 20%;">HNr<br>1867</td>
    <td style="width: 20%;">Eigentümer 1867</td>
    <td style="width: 20%;">HNr<br>1892</td>
    <td style="width: 30%;">Eigentümer 1840</td>
    <td style="width: 20%;">HNr<br>1810</td>
    <td style="width: 30%;">Eigentümer 1810</td>
</tr>
<?php
while($row = $ergebnis1->fetch_assoc())
{
    echo "<tr><td style=width: 6%;>$row[hausnr1913]</td>";
    echo "<td style=width; 3%;>$row[plannr]</td>";
    echo "<td style=width; 3%;>$row[schluessel1913]</td>";
    echo "<td style=width; 3%;>$row[schluessel1918]</td>";
    echo "<td style=width: 10%;>$row[strassenname1918]
$row[hausnrneu]</td>";
    echo "<td style=width: 20%;>$row[besitzer1913]</td>";
    echo "<td style=width: 20%;>$row[hausnrumschr]</td>";
    echo "<td style=width: 20%;>$row[besitzerumschr]</td>";
    echo "<td style=width: 20%;>$row[hausnr1892]</td>";
    echo "<td style=width: 20%;>$row[besitzer1892]</td>";
    echo "<td style=width: 20%;>$row[hausnr1867]</td>";
    echo "<td style=width: 20%;>$row[besitzer1867]</td>";
    echo "<td style=width: 20%;>$row[hausnr1840]</td>";
    echo "<td style=width: 20%;>$row[besitzer1840]</td>";
    echo "<td style=width: 30%;>$row[hausnr1810]</td>";
    echo "<td style=width: 20%;>$row[besitzer1810]</td></tr>";
}

```

```

    }
    mysqli_close($db);
  }
?>

</tbody>
</table>

<hr>
<center>
<a target="_blank" Href="index-kat.html">Übersicht Kataster</a>
</center>
<hr>
<small>Änderungsstand: 28-Feb-2019<br>
<a href=" ../Impressum.html">Heinz Wember</a></small>
</body>
</html>

```

Programm top.php

```

<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
  <style type="text/css">
    @import url("../FORMATE.CSS") screen;
  </style> <meta name="author" content="Heinz Wember Augsburg, php7, 09-Apr-2019">

</head>
<body>
<!-- web:picture:start --><!-- web:picture:stop -->

</body>
</html>

```

Programm uebersicht.php:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("FORMATE.CSS") screen;
  </style>
  <meta name="author" content="Heinz Wember Augsburg, hpp7, 28-Feb-2019">
  <title>DB Heinz Wember Dokumentation: Übersicht</title>
</head>
<body>
<br><center>
<h1>DB Heinz Wember Dokumentation: Übersicht</h1>
</center><br>
<?php
  define("UNLOCK",1);
  include "../db/zugriff-dbdoku.inc.php";
  $abfrage = "SELECT name, gruppe, active, sort, erstellt
    FROM uebersicht
    WHERE active ='j'
    ORDER BY gruppe, sort";
  $ergebnis = mysqli_query($db, $abfrage);
?>
  <br><h1>Übersicht</h1><br>
  <table style="width: 100%; text-align: left; margin-left: left; margin-right: auto;" border="1" cellpadding="2" cellspacing="2">
  <caption></caption><tbody>
    <tr>
      <td style="width: 40%;">Name der Dokumentation</td>
      <td style="width: 20%;">Gruppe</td>
      <td style="width: 05%;">Sortierung</td>
      <td style="width: 10%;">Änderungsdatum</td>
    </tr>

  <?php
  while($row = $ergebnis->fetch_assoc())
  {
    echo "<tr> ";
    echo "<td style=width: 40%;> $row[name]</td>";
    echo "<td style=width: 20%;> $row[gruppe]</td>";
    echo "<td style=width: 05%;> $row[sort]</td>";
    echo "<td style=width: 10%;> $row[erstellt]</td></tr>";
  }
  mysqli_close($db);
?>
</tbody></table>
<hr><center>
<a target="_blank" href="index.html">Übersicht Dokumentationen</a>
</center>
<hr>
<small>Änderungsstand des Programms: 2018-Feb-2019<br></small>
</body></html>
```

Programm DB öffnen

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
  <meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
  <title>Connect to MySQL-DB</title>
</head>
<body>

<?php
  define ( 'MYSQL_HOST', 'localhost' );
  define ( 'MYSQL_BENUTZER', 'dbo99999999' );
  define ( 'MYSQL_KENNWORT', 'falsch' );
  define ( 'MYSQL_DATENBANK', 'db99999999' );

  $db = mysqli_connect (MYSQL_HOST, MYSQL_BENUTZER, MYSQL_KENNWORT, MYSQL_DA-
TENBANK);

  if ( $db )
  {
  }
  else
  {
    echo "Fehler: konnte nicht mit MySQL verbinden." . PHP_EOL;
    echo "Debug-Fehlernummer: " . mysqli_connect_errno() . PHP_EOL;
    echo "Debug-Fehlermeldung: " . mysqli_connect_error() . PHP_EOL;
    exit;
  }
?>
</body>
</html>
```

Ausgabe Datum formatiert:

Kirchenchor Kriegshaber Dreifaltigkeit

Chortermine in der Zukunft

Datum	Zeit	Kirche	Programm
Do, 09-Apr-2020	19:00	Dreifaltigkeit	Gründonnerstag
Fr, 10-Apr-2020	15:00	Dreifaltigkeit	Karfreitag
Mo, 13-Apr-2020	10:00	Dreifaltigkeit	Ostermontag Haydn: Orgelsolomesse B-dur für Soli, Chor und Orchester; Mozart: Laudate dominum für Solo, Chor und Orchester;
Do, 01-Jan-2099		Änderungsstand:	db-Tabelle termine: 16-Feb-2020

[Übersicht Kirchenchor](#)

Änderungsstand: 17-Jun-2019

Ausgabe Datum normal:

Kirchenchor Kriegshaber Dreifaltigkeit

alle Chortermine in der Zukunft und Vergangenheit

Datum	Zeit	Kirche	Programm
Do, 01.01.2099		Änderungsstand:	db-Tabelle termine: 16-Feb-2020
Mo, 13.04.2020	10:00	Dreifaltigkeit	Ostermontag Haydn: Orgelsolomesse B-dur für Soli, Chor und Orchester; Mozart: Laudate dominum für Solo, Chor und Orchester;
Fr, 10.04.2020	15:00	Dreifaltigkeit	Karfreitag
Do, 09.04.2020	19:00	Dreifaltigkeit	Gründonnerstag
Mo, 06.01.2020	09:30	Thaddäus	Dreikönig: Mozart Missa brevis in B KV 275 für Sol, Chor und Orchester Gjeilo: Northern Lights (Pulchra es amica mea)
Do, 26.12.2019	10:00	Dreifaltigkeit	Weihnachten Karl Kempfer Pastoralmesse in G für Soli, Chor und Orchester; Karl Kempfer: Hodie_Christus_natus_est Gjeilo: Northern Lights (Pulchra es amica mea)
So, 20.10.2019	10:00	Dreifaltigkeit	Kirchweih: Mozart Missa brevis in B KV 275 für Sol, Chor und Orchester
So, 13.10.2019	17:00	Dreifaltigkeit	<u>Kirchenkonzert 2019</u> Programm des gemischten Kirchenchores: Liedblatt 875, Ave Maria von Lonquich, Ave Maria von Rachmaninov, Northern Lights von Gjeilo, Ubi caritas von Snyder, Regina Coeli von Lotti, Ave verum corpus von Elgar, Blei bei uns Herr von Monk/L'Estrange, Ave Maria von Arcadelt; Frauen des Kirchenchores: Laudate pueri von Mendelssohn; Weitere Programmteile von den Chören Gaudeamus und Kinderchor
So, 16.06.2019	10:00	Dreifaltigkeit	Patrozinium: Mozart Missa brevis für Soli Chor und Orchester "Spatzenmesse" KV 220 C-dur; Knut Nystedt: Laudate dominum Mozart: Tantum ergo für Solo, Chor und Orchester KV 142 B-dur;
Mo, 22.04.2019	10:00	Dreifaltigkeit	Ostermontag Mozart: Krönungsmesse C-dur, KV 317 für Soli, Chor und Orchester; Mozart: Soli für S und T: Sub tuum praesidium F-dur KV 198; Händel: Halleluja aus dem Messias für Soli, Chor und Orchester;
Fr, 19.04.2019	15:00	Dreifaltigkeit	Karfreitag Chor und Volksgesang: Herr, auf dich vertraue ich (GL665); Chor und Volksgesang: Lass mich deine Leiden singen (GL733,1) Chor: J. S. Bach: O Haupt voll Blut und Wunden; Andreas Wilscher: Domine exaudi orationem meam Rihards Dubra: O crux ave Spiritual: Herr, ich möcht dein Leiden fassen; Chor und Volksgesang: Selig, wem Christum auf dem Weg begegnet (GL 275);

file:///F:/Wember/Heinz/BuchMySQL/Auswertungen/tmp/Chorterminealle.html[26.03.2020 04:46:49]

Ausgabe Apotheker:

Kataster Kriegshaber, Auswahl Text in den Katastern

Kataster Kriegshaber, Auswahl Text in den Katastern

Auswahl für %Schärtl%Apotheker% : Anzahl der Sätze: 2

Plannummer	Besitzer
077	Jetzige Adresse: StrNrheute: 1010 Ulmer Straße 209; Hausnummer 1913: 091
077	Kataster 1810: H#
077	Kataster 1840]: H#033 Bachmann Seligmann Handelsmann P#077, 1/1 Haus,
077	Kataster 1854: H#033 neue H#074: Sebastian Spieß P#077 WH etc: 0,09 Tgw Gemeinderecht: ganzen N. P#078: Garten: 0,02 Tgw ... vom 26-Jul-1844 und 17-Jul-1845 von Martin Bachmann um 950 fl erkauf Umschr: KS#061 1/2
077	Kataster 1867: H#074 Schärtl Georg P#077: Wohnhaus mit Hofraum, 0,09 Tgw Gemeinderecht: ganzer N. Lit C: P#078: Garten, 0,02 Tgw ... vom 12-Jun-1861 von Johann Spieß um 3.000 fl erkauf einschließlich der auswärtigen Besitzungen Umschr KS#180 1/2
077	Kataster 1892: H#091 Schärtl Georg junior Apotheker Ulmer Str. 209 P#077: Wohnhaus mit Apotheke und Hofraum P#078: Baumgarten mit Pumpbrunnen P#079: Gras- und Baumgarten nebst Sommerhäuschen der Gesamtbesitz theils nach Umschreibverzeichnis ... vom 09-Sep-1868 von Georg Schärtl senior übernommen, theils nach ... 08-Mar-1873 von Raphael Thraenl? Bauer und theils nach ... vom 11-Jan-1887 von der Gemeinde erkauf Summe 0,072 ha GrSt 3,70 M Umschr KS#119 1/2
	Umschreib-Kataster 1892: H#091 Schärtl Georg junior Apotheker Ulmer Str. 209 02-Apr-1892: Nothaas Max Apotheker und Rosina geb. Buchloh Kauf 02-Mai-1933: Buchloh Heinrich Erbfolge 19-Mai-1934: Kauf von Steppich Jakob H#004 Kriegshaberstr. um 600 M 19-Mai-1936: Messung Grundabtretung 0,013 ha an die Stadt Augsburg IV. VJ 1941: Abgang P#077, P#078 und P#079 mit 0,0590 ha Rest 0,0100 ha: Um- und Anbau im August 1934 Messungs... dagegen Zugang: P#077 Wohnhaus mit Apotheke, Laden Einfahrt, Hof und Garten

file:///F:/Wember/Heinz/BuchMySQL/Auswertungen/tmp/SchaertlApotheker.html[26.03.2020 04:45:55]

077	<p>Gebäude H#209 Ulmer Str. mit 0,0590 ha Stand 0,0690 ha P#078 und P#079 fallen weg</p> <p>IV VJ 1944: P#077 und P#079 1/2 0,0690 Rest 0 dagegen Zugang: P#077/1 Wohnhaus mit Apotheke und Laden, Hofraum und Garten 0,0670 ha wegen Neumessung 20 qm Minderung II. VJ 1962</p> <p>auf Ableben Buchloh Heinrich u. Buchloh ... geb. Aumberger (?) Apothekenbesitzerswitwe allein letzter Eintrag</p>
077	Adressbuch 1913: H#091 Nothaas Josina, Apothekerswitwe (1913); Seitz für Nothaas (1917)
077	Katasterseite:
077	Repertorium/Messverzeichnis:
078	Jetzige Adresse: StrNrheute: 1010 Ulmer Straße 209; Hausnummer 1913:
078	Kataster 1810: H#
078	Kataster 1840]: H# Bachmann Seligmann Handelsmann
078	Kataster 1854: H#
078	Kataster 1867: H# Schärfl Georg P#077, P#078 0,02 Tgw am 12-Jun-1861 von Johann Spieß erkaufte
078	Kataster 1892: H# Schärfl Georg Apotheker P#077, dazu P#078 und P#079
078	Umschreib-Kataster 1892: H#
078	Adressbuch 1913: H# 1918 zu Apotheke Nothaas P#077 Ulmer Str. 209 P#078 Baumgarten und Brunnen P#079 Gras- u Baumgarten
078	Katasterseite:
078	Repertorium/Messverzeichnis:

[Katasterdaten unter Passwortschutz](#)

Änderungsstand: 24-Apr-2019

[Heinz Wember](#)

Ausgabe Bevölkerungsstatistik:

Summe der Entfernung vom Geburtsort der Männer - 001 bis 020 km

Kreis	Einzelentfernung	Anzahl Männer	Männer mal Entfernung
Augsburg-Kreis	20	236	4720
Augsburg-Stadt	5	68	340
Friedberg	15	18	270
Summe		Anzahl Männer	Männer mal Entfernung
Summe	Entfernung 1 bis 20 km	322	5330

[Übersicht Kirchenbücher](#)

Änderungsstand: 25-Feb-2019

[Heinz Wember](#)

Ausgabe Katband1:

Staatsarchiv Augsburg: Katasterbände, Sort: Steuergemeinde, Gruppe, Typ

Staatsarchiv Augsburg: Quellenbände, Sort: Steuergemeinde, Gruppe, Typ

Kataster: Gruppe 1 bis 13; Gruppe = 0 andere Quellen (Grundbuch, Neues Grundbuch, Gefälle)

ID	Rentamt	Steuergemeinde	Steuerdistrikt	Land-Amtsgericht	Typ	Gruppe	Abk	Langtext zur Abk
105	zzzzz	Änderungsstand katkopf: 15- Dez-2018			zzzzz	0		
162	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askriggefäll	0	gefäll	Gefällbuch des königlich bayerischen Rentamtes Göggingen, Steuer-Districts Kriegshaber bestehend aus den Ortschaften Kriegshaber, 1824
141	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askrigrundb	0	grundb	Grundbuch
160	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askringbanh	0	ngbanh	Grundbuch über die von der St. Adelgundis Pfarrkirchenstiftung in Anhausen an die Ablösungs- Cassa des Staates abgetretenenen Grundrenten in den Steuergemeinden Anhausen, Kriegshaber und Westheim
155	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askringbhosp	0	ngbhosp	Neues Grundbuch über die von der paritätischen Stiftung in Augsburg an die Ablösungs-Cassa des Staates überwiesenen Geld- und Getreid-Gefälle resp. regulierten Bodenzinsen hieraus in den Steuergemeinden 14 Gemeinden, darunter Kriegshaber
156	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askringbhosp	0	ngbhosp	Neues Grundbuch über die von der paritätischen Stiftung in Augsburg an die Ablösungs-Cassa des Staates überwiesenen Geld- und Getreid-Gefälle resp. regulierten Bodenzinsen hieraus in den Steuergemeinden 14 Gemeinden, darunter Kriegshaber
159	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askringbkriegsh	0	ngbkriegsh	Neues Grundbuch des kgl. Rentamtes Göggingen über die bisherigen fixierten Grundgefälle des Staates und deren Umwandlung in ablösbare Boden-Zinse für die Gemeinde Kriegshaber
158	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askringbmartin	0	ngbmartin	Neues Grundbuch über die regulierten Bodenzinse aus den von der St. Martins- Stiftung in Augsburg an die Ablösungs-Cassa des Staates überwiesenen Grund- und Zehentrenten in den Steuergemeinden 6 Steuergemeinden, darunter Kriegshaber

Ausgabe Katband2:

Staatsarchiv Augsburg: Katasterbände, Sort: Steuergemeinde, Gruppe, Typ

161	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askringboberh	0	ngboberh	Grundbuch der Ablösungs-Cassa Pfarrei Oberhausen in der Steuergemeinde Kriegshaber
157	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askringbrehl	0	ngbrehl	Grundbuch über die von dem Freyherrn von Rehlingen in Hainhofen für die Ablösungs-Casse des Staates übernommenen und in den Bodenzins umgewandelten Geld- und Naturalgrundgefälle in den Steuergemeinden 6 Steuergemeinden, darunter Kriegshaber
5	Augsburg-Land	Kriegshaber	Augsburg-Land	Landgericht Augsburg	alkribf	1	bf	Besitzfassion, angelegt
6	Augsburg-Land	Kriegshaber	Augsburg-Land	Landgericht Augsburg	alkribf	1	bf	Besitzfassion, angelegt
7	Augsburg-Land	Kriegshaber	Augsburg-Land	Landgericht Augsburg	alkrihr	2	hr	Häuser- und Rustikalsteuerkataster, angelegt
1	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askrigruk	4	gruk	Grundsteuerurkataster, abgeschlossen
2	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askriukg	6	ukg	Umschreibgrundsteuerkataster gebunden
11	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askriuugh	7	uugh	Umschreibhefte zum Umschreibgrundsteuerkataster von bis
12	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askriuugh	7	uugh	Umschreibhefte zum Umschreibgrundsteuerkataster von bis
13	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askrigr1	8	gr1	Grundsteuerkataster 1. Renovation
14	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askrigr2	8	gr2	Grundsteuerkataster 2. Renovation
15	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askrigr3	8	gr3	Grundsteuerkataster 3. Renovation
16	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askrifl1r	9	fl1r	Flächenrepertorium zur 1. Renovation, Abschluss
17	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askrifl2r	9	fl2r	Flächenrepertorium zur 2. Renovation, Abschluss
19	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askrifl3r	9	fl3r	Flächenrepertorium zur 3. Renovation, Abschluss
18	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askrifl3r	9	fl3r	Flächenrepertorium zur 3. Renovation, Abschluss
20	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askriuh1g	10	uh1g	Umschreibhefte zum 1. renovierten Grundsteuerkataster
21	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askriuh1g	10	uh1g	Umschreibhefte zum 1. renovierten Grundsteuerkataster
22	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askriuh2g	10	uh2g	Umschreibhefte zum 2. renovierten Grundsteuerkataster
24	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askriuh2g	10	uh2g	Umschreibhefte zum 2. renovierten Grundsteuerkataster
25	Augsburg-Stadt	Kriegshaber	Augsburg-Land	Landgericht Augsburg	askriuh2g	10	uh2g	Umschreibhefte zum 2. renovierten Grundsteuerkataster

Quelle Staatsarchiv:

Regiment: Augsburg - Stadt
 Steuergemeinde: Kriegshaber
 Steuerdistrikt:
 vordem Augsburg - Land

Landgericht/Amtsgericht:
 vordem: Landgericht Augsburg

1 Besitzfassungen A-Land angelegt: <i>bf</i> <i>1809</i>	Nr. 32 I Pass. 1-173	32 II 184-334	
2 Häuser- und Rustikalsteuer- A-Land kataster, angelegt: <i>1810</i>	Nr. 12 <i>AL</i> Besl. Nr. 1-649		
3 Umschreibbuch zu 2 von bis	Nr. 12 <i>649</i>		
4 Grundsteuerkataster <i>gr 1</i> Urkataster abgeschlossen: <i>u. kg. 1840</i>	Nr. 48 Ks. 1-335 Hs. Nr. 1-Ende	N Ks. Hs. Nr.	N Ks. Hs. Nr.
5 Flächenrepertorium zu 4	Nr. <i>1840</i>		
6 Umschreibgrundsteuer- kataster, gebunden u. kg. von 1832 bis 1842 <i>u. kg.</i>	Nr. 77		
7 Umschreibhefte zum Um- schreibgrundsteuerkataster von 1842 bis 1853 <i>u. kg. h.</i>	Nr. 91 I Ks. 3-200 Hs. Nr. 1-70	Nr. 91 II Ks. 202-336 Hs. Nr. 71-Ende	Nr. Ks. Hs. Nr.
8 Grundsteuerkataster <i>gr 1</i> erste Renovation vom Jahre <i>1854</i>	Nr. 108 I Ks. 1-334 Hs. Nr. 1-Ende	Nr. Ks. Hs. Nr.	Nr. Ks. Hs. Nr.
Grundsteuerkataster <i>gr 2</i> zweite Renovation vom Jahre <i>1867</i>	Nr. 108 II Ks. 1- 334 508 Hs. Nr. 1-Ende	N Ks. Hs. Nr.	N Ks. Hs. Nr.
Grundsteuerkataster <i>gr 3</i> dritte Renovation vom Jahre <i>1892</i>	Nr. Ks. 1-1120 Hs. Nr. 1-Ende	N Ks. Hs. Nr.	N Ks. Hs. Nr.
9 Flächenrepertorium zur 1. Renovation, Abschluß <i>1854</i>	Nr. 177 I		
Flächenrepertorium zur 2. Renovation, Abschluß <i>1867</i>	Nr. 177 II		
Flächenrepertorium zur 3. Renovation, Abschluß <i>1892</i>	Nr. 1892-1940	1940-1961	
<i>1819</i> Umschreibhefte zum ersten renovierten Grundsteuer- kataster, von 1854 bis 1867	Nr. 141 I Ks. 1-170 Hs. Nr. 1-99	Nr. 141 II Ks. 172-334 Hs. Nr. 101-Ende	Nr. Ks. Hs. Nr.
<i>1829</i> Umschreibhefte zum zweiten renovierten Grundsteuer- kataster, von 1867 bis 1892	Nr. 141 III Ks. 2-170 Hs. Nr. 1-74	Nr. 141 IV Ks. 181-340 Hs. Nr. 75-155	Nr. 141 V Ks. 341-508 Hs. Nr. 74-Ende
<i>1839</i> Umschreibhefte zum dritten renovierten Grundsteuer- kataster, von 1892 bis 1961	Nr. Ks. 2-110 Hs. Nr. 1-86	Nr. Ks. 112-240 Hs. Nr. 87-158	Nr. <i>1839-1861</i> Ks. 241-350 Hs. Nr. 159-259
11 Anmeldeprotokolle <i>ap</i> von 1832 bis 1862	Nr. 63 I 1832-1855	Nr. 63 II 1856-1862	Nr.
12 Umschreibverzeichnisse <i>uv</i> von 1862 bis 1909	Nr. 156	Nr.	Nr.
13 Messungsverzeichnisse <i>mv</i> von 1909 bis 1964	Nr. <i>siehe Verzeichnisse</i>	Nr.	Nr.
14 Gewerbesteuerkataster	Nr. Jhrg.	Nr. Jhrg.	Nr. Jhrg.
15 Umschreibbuch zu 14 von bis	Nr.	Nr.	Nr.

Ausgabe Navigation:

Connect to MySQL-DB

00	00	j	Hauptmenü	Hauptmenü
00	05	j	Strassennamen und Hausnummern	Straßennamen und Hausnummern
00	06	j	Strassen nach Straßenschlüssel sortiert	Strassen nach Straßenschlüssel sortiert
00	07	j	Geschichte der Hausnummern	Geschichte der Hausnummern
00	08	j	Strassen mit Strassenschlüssel sortiert nach Namen	Strassen mit Straßenschlüssel sortiert nach Namen
00	10	j	Hausnr alt zu Hausnr heutiger Adressierung	Hausnr alt zu Hausnr heutiger Adressierung
00	11	j	Hausnr alt zu Hausnr heutiger Adressierung, nur Grundstücke mit Adresse	Hausnr alt zu Hausnr heutiger Adressierung, nur Grundstücke mit Adresse
00	14	j	Hausnr alt zu Hausnr neu nach altem Strassennamen sortiert	Hausnr alt zu Hausnr neu nach altem Straßennamen sortiert
00	16	j	Hausnr alt zu Adressierung von 1917	Hausnr alt zu Adressierung von 1917
00	17	j	Hausnr neu zu Hausnr alt, 1917	Hausnr neu zu Hausnr alt 1917
00	18	j	Crossreference nach Eigentüemer	Crossreference nach Eigentümer
00	19	j	alle Häuser heute	alle Häuser heute
00	20	j	Auswertungen fürs Buch	Auswertungen fürs Buch
00	21	j	alle Häuser heute	alle Häuser heute
00	25	j	Details zu den Straßen von Kriegshaber	Details zu den Straßen von Kriegshaber
00	30	j	Kataster Auswertungen	Kataster-Auswertungen
00	34	j	Ablösung der Grundlasten	Ablösung der Grundlasten
00	40	j	Kirchenbücher	Kirchenbücher
00	50	j	Geschichte Immobilien der Juden	Gesch Immobilien der Juden
00	52	j	Geschichte einiger Häuser, u.a. Kirche, Synagoge	Geschichte einiger Häuser, u.a. Kirche, Synagoge
00	54	j	Fotos Kriegshaber	Fotos Kriegshaber
00	57	j	Strassennamen in Kriegshaber	Straßennamen in Kriegshaber
00	58	j	Grenzen von Kriegshaber seit 1810	Grenzen von Kriegshaber seit 1810
00	59	J	Status 1920	Status 1920
00	60	j	Strassennamen 1938/heute	Straßennamen 1938/heute
00	61	j	Strassen in Augsburg nach Namen	Straßen in Augsburg nach Namen
00	62	j	Strassen in Augsburg nach Bezirknr	Straßen in Augsburg nach Bezirknr
00	64	j	Quellen für Kriegshaber	Quellen für Kriegshaber
00	65	j	Quellen StAA	Quellen StAA
00	66	j	Quellen Bistums-AA	Quellen Bistums-AA
00	67	j	Quellen StadtAA	Quellen StadtAA
00	68	j	Quellen Fotos/Bilder Übersicht	Quellen Fotos/Bilder Übersicht
00	69	j	Quellen Fotos/Bilder StadtAA	Quellen Fotos/Bilder StadtAA
00	70	j	Quellen Heimatforschung	Quellen Heimatforschung
00	71	j	Seminar Kriegshaber	Seminar Kriegshaber
00	71	j	sonstige Quellen	sonstige Quellen

Ausgabe Textgestaltung:

Gestaltung

-- Anhang zu Buch Die Datenbank MySQL als Hilfsmittel für die -- Darstellung geschichtlicher Themen: Datei
gestaltung.txt

Textgestaltung mit html

Dies ist eine Zeile zwischen h1

Dies ist eine Zeile zwischen h2

Dies ist eine Zeile zwischen h3

Dies ist eine Zeile zwischen h4

Dies ist eine Zeile zwischen h5

dies ist ein Text zwischen bold

dies ist ein Text zwischen strong

alle html-Steuerzeichen sind mit spitzer Klammer auf und spitzer Klammer zu gekennzeichnet. Die Zeichen spitze Klammer auf und zu dürfen also nicht als Textzeichen verwendet werden. Die Zeichen spitze Klammer stehen auf der Tastatur links neben y. Der Text wird also zwischen diesen Steuerzeichen eingeschlossen:

im Beispiel bold: es kommt spitze Klammer auf dann das Zeichen b dann spitze Klammer zu; nun folgt der Text, der entsprechend hervorgehoben ist, dann das Endezeichen, das wie das Anfangzeichen ist, jedoch zuvor einen Schrägstrich hat, also spitze Klammer auf Schrägstrich und dann das eigentliche Steuerzeichen (in diesem Beispiel b) und spitze Klammer zu.

Genauereres siehe im [Handbuch](#)

Wie nun die einzelnen html-Steuerzeichen interpretiert werden, ist in der Formatdatei (bei mir FORMAT.CSS) definiert, die im Root-Verzeichnis steht. Man kann natürlich für unterschiedliche Unterverzeichnisse auch unterschiedliche Formatdateien nehmen, was ich aber nicht mache, weil ich auf ein einheitliches Erscheinungsbild achte.

[Datei FORMATE.CSS](#)

Diese Homepage ist so designed, dass bei allen Fenstergrößen, egal ob 800 x 600, 1024 x 768 oder auch 1680 x 1050 Pixel die Bildschirmgröße optimal genutzt wird. Bei kleinen Fenstergrößen erscheint das Navigationsfenster natürlich nicht vollständig auf dem Bildschirm. Dann kommt die Windows-Laufleiste. Außerdem kann man die Breite des Navigationsfenster auf Kosten des großen rechten Fensters mittels linker Maustaste am Fensterrand vergrößern.

Im Großen und Ganzen sind die gängigen Browser in der Interpretation der Steuerzeichen alle gleich. Die Farben der

Gestaltung

Schriften (bei h1 bis h5 und bold) sind nur sichtbar bei den Browsern Mozilla Firefox, Opera und Google Chrom, nicht jedoch bei Microsoft Internet Explorer 8.0 in Verbindung Windows Vista. Bei Google Chrom ist mir aufgefallen, dass das Verändern der Breite des Fensters Navigation nicht funktioniert.

Änderungsstand gestaltung: 28-Jan-2014 Upd 25-Mar-2020

FotoDoku connect auf die MySQL-DB: php rel. 7.2

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
  <meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
  <title>Musterhomepage mit PHP und MySQL TEST</title>
  <meta name="author" content="php7: Heinz Wember Augsburg">
  <!-- Heinz Wember 30-Dez-2018 -->
</head>
<body>

<?php
define ( 'MYSQL_HOST', 'localhost' );
define ( 'MYSQL_BENUTZER', 'Benutzer' );
define ( 'MYSQL_KENNWORT', 'passwort' );
define ( 'MYSQL_DATENBANK', 'DBName' );

$db = mysqli_connect(MYSQL_HOST, MYSQL_BENUTZER, MYSQL_KENNWORT, MYSQL_DA-
TENBANK);

if ($db)
{
}
else
{
  echo "Fehler: konnte nicht mit MySQL verbinden." . PHP_EOL;
  echo "Debug-Fehlernummer: " . mysqli_connect_errno() . PHP_EOL;
  echo "Debug-Fehlermeldung: " . mysqli_connect_error() . PHP_EOL;
  exit;
}
?>
</body>
</html>
```

FotoDoku fixe Spaltenbreite index.html php rel. 7.2

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("FORMATE.CSS") screen;
    body { font-family:Arial,sans-serif; } body { background-color: #FFFF00; } // #FFFF00 gelb; #98FB98
hellgrün; #40E0D0 türkis; #F4A460 sandbraun;
    a:link { color:#000000; text-decoration:underline;}
    a:visited { color:#FF0000; text-decoration:underline; }
    a:focus { color:#00AA00; background-color:#FFFF77;}
    a:hover { color:#EE0000; text-decoration:none; background-color:#FFFF99; }
    a:active { color:#0000EE; background-color:#FFFF99;}
  <meta name="DC.Language" content="de">
  <meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
  </style>
  <meta name="author" content="php Rel 7.xxxx Heinz Wember Augsburg, relative Spaltenbreite, 24-Jan-
2019">
</head>

<frameset rows="210px,800px">
  <frameset cols="210px,800px">
    <!-- Zwei Frames, deren Inhalt hier bestimmt wird. -->
    <frame name="logo" src="logo-fix.php" scrolling="no" frameborder="no" >
    <frame name="top" src="top-fix.php" scrolling="no" frameborder="no" >
  </frameset>
  <frameset cols="210px,800px">
    <!-- Noch mal zwei Frames, deren Inhalt hier bestimmt wird. -->
    <frame name="navigation" src="navigation02.php" scrolling="auto" frameborder="no" >
    <frame name="content" src="fotosnachgruppeundjahrfix.php" scrolling="auto" frameborder="no">
  </frameset>
</frameset>

<body>
Sie befinden sich auf der Homepage MySQL-FotoDoku
erstellt von Heinz Wember<br>
24-Jan-2019
</body>
</noframes>
</body>
</html>
```

FotoDoku relative Spaltenbreite navigation php rel. 7.2

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
<style type="text/css">
  @import url("FORMATE.CSS") screen;
  body { font-family:Arial,sans-serif; }
  a:link { color:#000000; text-decoration:underline;}
  a:visited { color:#FF0000; text-decoration:underline; }
  a:focus { color:#00AA00; background-color:#80FF80;}
  a:hover { color:#EE0000; text-decoration:none; background-color:#FFFF99; }
  a:active { color:#0000EE; background-color:#FFFF99;}
<meta name="DC.Language" content="de">
</style>
<meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
<meta name="author" content="php Rel 7.xxx Heinz Wember Augsburg, relative Breite, 24-Jan-2019">
</head>
<br>
<left>
<small>
<table width=15%; text-align: left; margin-left: auto; margin-right: auto;" border="1" cellpadding="2" cell-
spacing="2">
  <tbody>
  <?php
define("UNLOCK",1);
include "../db/zugriff-db-fotodoku.inc.php";
if ($db)
{
}
else
{
  echo "Fehler: konnte nicht mit MySQL verbinden." . PHP_EOL;
  echo "Debug-Fehlernummer: " . mysqli_connect_errno() . PHP_EOL;
  echo "Debug-Fehlermeldung: " . mysqli_connect_error() . PHP_EOL;
  exit;
}
$sql = "SELECT `gruppe` , `ugruppe` , `titel`
FROM `navigation`
WHERE `gruppe` = '00' AND aktiv = 'j'
AND `KZ` < 2
ORDER BY `ugruppe`";
$ergebnis = mysqli_query($db, $sql);
while($row = $ergebnis->fetch_assoc())
{
  echo "<tr> $row[titel] </tr><br>";
}
mysqli_close($db);
?>
</tbody></table></small>
</body></html>
```

FotoDoku relative Spaltenbreite php rel. 7.2

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("FORMATE.CSS") screen;
  </style>
  <meta name="DC.Language" content="de">
  <meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
  <meta name="author" content="php Rel. 7.xxxx Heinz Wember Augsburg, 22-Jan-2019">
  <title>Dokumentation Fotos</title>
</head>
<body><br><h1>Alle Sätze sortiert nach Ablageort</h1>
<br>
<?php
  define("UNLOCK",1);
  include "../db/zugriff-db-fotodoku.inc.php";
  $sql = "SELECT fotos.nummer AS nr, fotos.abk AS abk, gruppe.beschreibung AS beschr, fotos.jahr AS
jahr,
        fotos.personen AS personen, fotos.motiv AS motiv, minifoto
        FROM fotos
        LEFT JOIN gruppe ON gruppe.abk = fotos.abk
        ORDER by nr";
  $ergebnis = mysqli_query($db, $sql);
  ?>
<table style="width: 85%; text-align: left; margin-left: left; margin-right: auto;" border="1" cellpadding="2"
cellspacing="2">
  <caption></caption><tbody>
    <tr>
      <td style="width: 20%;>Fotonummer<br></td>
      <td style="width: 05%;>Abk</td>
      <td style="width: 10%;>Beschreibung</td>
      <td style="width: 05%;>Jahr</td>
      <td style="width: 10%;>Personen</td>
      <td style="width: 40%;>Motiv</td>
      <td style="width: 10%;>Foto</td>
    </tr>
<?php while($row = $ergebnis->fetch_assoc())
  {
    echo "<tr><td style=width: 20%; font-family: Arial; font-weight: bold;>$row[nr]</td>";
    echo "<td style=width: 05%; font-family: Arial; font-weight: bold;>$row[abk]</td>";
    echo "<td style=width: 10%; font-family: Arial; font-weight: bold;>$row[beschr]</td>";
    echo "<td style=width: 05%; font-family: Arial; font-weight: bold;>$row[jahr]</td>";
    echo "<td style=width: 10%; font-family: Arial; font-weight: bold;>$row[personen]</td>";
    echo "<td style=width: 40%; font-family: Arial; font-weight: bold;>$row[motiv]</td>";
    echo "<td style=width: 10%; font-family: Arial; font-weight: bold;>$row[minifoto]</td></tr>";
  }
  mysqli_close($db);
  ?>
  </tbody></table><hr><center>
<a target=_blank href="index.html">Übersicht</a></center><hr><left>
<small>&Auml;nderungsstand: 22-Jan-2019<br>
<a href="../Impressum.html">Heinz Wember</a></small><br></body></html>
```

FotoDoku relative Spaltenbreite php Rel. 7.2: Suche nach Motiv

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("FORMATE.CSS") screen;
  </style>
  <meta name="DC.Language" content="de">
  <meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
  <meta name="author" content="php Rel. 7.xxxx Heinz Wember Augsburg, 22-Jan-2019">
  <title>Fotos suche Motiv</title>
</head>
<body>
<br>

<h1>Fotos suche Motiv</h1>

<left>
<br>
<?php

  if (!$_POST['gesendet'] && !$_POST['motiv'])
  {
?>
  <form method="POST" action="<?php echo $PHP_SELF ?>">
  <p>Motiv z.B. %Zugspitze% oder %Matterhorn% <br>wenn nur % eingegeben wird, kommen alle Sätze":
    <input type="text" name="motiv" size="30">
  </p>
  <input type="submit" name="Submit" value="Submit">
  <input type="hidden" name="gesendet" value="1">
  </form>

<?php

  }

else {

  $motiv=$_POST[motiv];
  echo "Auswahl für $motiv";
  define("UNLOCK",1);
  include "../db/zugriff-db-fotodoku.inc.php";

  $sql = "SELECT COUNT(*)
          FROM fotos
          WHERE motiv LIKE '$motiv'";

  $ergebnis = mysqli_query($db, $sql);
  $anzahl = mysqli_fetch_array($ergebnis);
  echo " : Anzahl der Sätze: $anzahl[0]<BR><BR>"; // Gibt die Anzahl aus
  mysqli_close($db);
```

```

define("UNLOCK",1);
include "../db/zugriff-db-fotodoku.inc.php";
$abfrage2 = "SELECT fotos.nummer AS nr, fotos.abk AS abk, gruppe.beschreibung AS beschr, fotos.jahr
AS jahr,
    fotos.personen AS personen, fotos.motiv AS motiv, fotos.minifoto
FROM fotos
LEFT JOIN gruppe ON gruppe.abk = fotos.abk
WHERE motiv LIKE '$motiv'
ORDER by fotos.abk, fotos.jahr";

$ergebnis2 = mysqli_query($db, $abfrage2);
?>
<table style="width: 85%; text-align: left; margin-left: left; margin-right: auto;" border="1" cellpadding="2"
cellspacing="2">
<caption></caption><tbody>
<tr>
<td style="width: 20%;>Fotonummer<br>
</td>
<td style="width: 05%;>Abk</td>
<td style="width: 10%;>Beschreibung</td>
<td style="width: 05%;>Jahr</td>
<td style="width: 10%;>Personen</td>
<td style="width: 30%;>Motiv</td>
<td style="width: 20%;>Foto</td>
</tr>

<?php while($row = mysqli_fetch_object($ergebnis2))
{
    echo "<tr><td style=width: 20%; font-family: Arial; font-weight: bold;>$row->nr</td>";
    echo "<td style=width: 05%; font-family: Arial; font-weight: bold;>$row->abk</td>";
    echo "<td style=width: 10%; font-family: Arial; font-weight: bold;>$row->beschr</td>";
    echo "<td style=width: 05%; font-family: Arial; font-weight: bold;>$row->jahr</td>";
    echo "<td style=width: 10%; font-family: Arial; font-weight: bold;>$row->personen</td>";
    echo "<td style=width: 30%; font-family: Arial; font-weight: bold;>$row->motiv</td>";
    echo "<td style=width: 20%; font-family: Arial; font-weight: bold;>$row->minifoto</td></tr>";
}
mysqli_close($db);
?>

</tbody>
</table>

<hr>
<center>
<a target=_blank href="index.html">Übersicht</a>
</center>
<hr>
<left>
<small>&Auml;nderungsstand: 22-Jan-2019<br>
<a href="../Impressum.html">Heinz Wember</a></small><br>
</html>

```


FotoDoku connect auf die MySQL-DB: php rel. 7.2

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
  <meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
  <title>Musterhomepage mit PHP und MySQL TEST</title>
  <meta name="author" content="php7: Heinz Wember Augsburg">
  <!-- Heinz Wember 30-Dez-2018 -->
</head>
<body>

<?php
define ( 'MYSQL_HOST', 'localhost' );
define ( 'MYSQL_BENUTZER', 'Benutzer' );
define ( 'MYSQL_KENNWORT', 'passwort' );
define ( 'MYSQL_DATENBANK', 'DBName' );

$db = mysqli_connect(MYSQL_HOST, MYSQL_BENUTZER, MYSQL_KENNWORT, MYSQL_DA-
TENBANK);

if ($db)
{
}
else
{
  echo "Fehler: konnte nicht mit MySQL verbinden." . PHP_EOL;
  echo "Debug-Fehlernummer: " . mysqli_connect_errno() . PHP_EOL;
  echo "Debug-Fehlermeldung: " . mysqli_connect_error() . PHP_EOL;
  exit;
}
?>
</body>
</html>
```

FotoDoku fixe Spaltenbreite index.html php rel. 7.2

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("FORMATE.CSS") screen;
    body { font-family:Arial,sans-serif; } body { background-color: #FFFF00; } // #FFFF00 gelb; #98FB98
hellgrün; #40E0D0 türkis; #F4A460 sandbraun;
    a:link { color:#000000; text-decoration:underline;}
    a:visited { color:#FF0000; text-decoration:underline; }
    a:focus { color:#00AA00; background-color:#FFFF77;}
    a:hover { color:#EE0000; text-decoration:none; background-color:#FFFF99; }
    a:active { color:#0000EE; background-color:#FFFF99;}
  <meta name="DC.Language" content="de">
  <meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
</style>
  <meta name="author" content="php Rel 7.xxxx Heinz Wember Augsburg, relative Spaltenbreite, 24-Jan-
2019">
</head>

<frameset rows="210px,800px">
  <frameset cols="210px,800px">
    <!-- Zwei Frames, deren Inhalt hier bestimmt wird. -->
    <frame name="logo" src="logo-fix.php" scrolling="no" frameborder="no" >
    <frame name="top" src="top-fix.php" scrolling="no" frameborder="no" >
  </frameset>
  <frameset cols="210px,800px">
    <!-- Noch mal zwei Frames, deren Inhalt hier bestimmt wird. -->
    <frame name="navigation" src="navigation02.php" scrolling="auto" frameborder="no" >
    <frame name="content" src="fotosnachgruppeundjahrfix.php" scrolling="auto" frameborder="no">
  </frameset>
</frameset>

<body>
Sie befinden sich auf der Homepage MySQL-FotoDoku
erstellt von Heinz Wember<br>
24-Jan-2019
</body>
</noframes>
</body>
</html>
```

FotoDoku relative Spaltenbreite index.html php rel. 7.2

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("FORMATE.CSS") screen;
    body { font-family:Arial,sans-serif; } body { background-color: #FFFF00; } // #FFFF00 gelb; #98FB98
hellgrün; #40E0D0 türkis; #F4A460 sandbraun;
    a:link { color:#000000; text-decoration:underline;}
    a:visited { color:#FF0000; text-decoration:underline; }
    a:focus { color:#00AA00; background-color:#FFFF77;}
    a:hover { color:#EE0000; text-decoration:none; background-color:#FFFF99; }
    a:active { color:#0000EE; background-color:#FFFF99;}
  <meta name="DC.Language" content="de">
  <meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
  </style>
  <meta name="author" content="php Rel 7.xxxx Heinz Wember Augsburg, relative Spaltenbreite, 24-Jan-
2019">
</head>

<frameset rows="90,*">
  <!-- Dadurch ergeben sich zwei Frame-Fenster, deren Inhalt hier bestimmt wird.
    Das erste Frame-Fenster wird normal definiert, anstelle des zweiten
    folgt jedoch das folgende untergeordnete Frameset: -->
  <frame name="top" src="../MySQL-rel-Fotodoku/top-rel.php" scrolling="no" frameborder="no">
  <frameset cols="15%,85%" rows=1200>
    <frame name="navigation" src="navigation01.php" scrolling="auto" frameborder="no" >
    <frame name="content" src="fotosnachgruppeundjahr.php" scrolling="auto" frameborder="no">
  </frameset>
</frameset>

<body>
Sie befinden sich auf der Homepage MySQL-FotoDoku
erstellt von Heinz Wember<br>
24-Jan-2019
</body>
</noframes>
</body>
</html>
```

FotoDoku relative Spaltenbreite php Rel. 7.2:Motivauswahl

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("FORMATE.CSS") screen;
  </style>
  <meta name="DC.Language" content="de">
  <meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
  <meta name="author" content="php Rel. 7.xxxx Heinz Wember Augsburg, 22-Jan-2019">
  <title>Fotos suche Motiv</title>
</head>
<body>
<br>

<h1>Fotos suche Motiv</h1>

<left>
<br>
<?php

  if (!$_POST['gesendet'] && !$_POST['motiv'])
  {
?>
  <form method="POST" action="<?php echo $PHP_SELF ?>">
  <p>Motiv z.B. %Zugspitze% oder %Matterhorn% <br>wenn nur % eingegeben wird, kommen alle Sätze":
  <input type="text" name="motiv" size="30">
  </p>
  <input type="submit" name="Submit" value="Submit">
  <input type="hidden" name="gesendet" value="1">
  </form>

<?php

  }

else {

  $motiv=$_POST[motiv];
  echo "Auswahl für $motiv";
  define("UNLOCK",1);
  include "../db/zugriff-db-fotodoku.inc.php";

  $sql = "SELECT COUNT(*)
          FROM fotos
          WHERE motiv LIKE '$motiv'";

  $ergebnis = mysqli_query($db, $sql);
  $anzahl = mysqli_fetch_array($ergebnis);
  echo " : Anzahl der Sätze: $anzahl[0]<BR><BR>"; // Gibt die Anzahl aus
  mysqli_close($db);
```

```

define("UNLOCK",1);
include "../db/zugriff-db-fotodoku.inc.php";
$abfrage2 = "SELECT fotos.nummer AS nr, fotos.abk AS abk, gruppe.beschreibung AS beschr, fotos.jahr
AS jahr,
    fotos.personen AS personen, fotos.motiv AS motiv, fotos.minifoto
FROM fotos
LEFT JOIN gruppe ON gruppe.abk = fotos.abk
WHERE motiv LIKE '$motiv'
ORDER by fotos.abk, fotos.jahr";

$ergebnis2 = mysqli_query($db, $abfrage2);
?>
<table style="width: 85%; text-align: left; margin-left: left; margin-right: auto;" border="1" cellpadding="2"
cellspacing="2">
<caption></caption><tbody>
<tr>
<td style="width: 20%;>Fotonummer<br>
</td>
<td style="width: 05%;>Abk</td>
<td style="width: 10%;>Beschreibung</td>
<td style="width: 05%;>Jahr</td>
<td style="width: 10%;>Personen</td>
<td style="width: 30%;>Motiv</td>
<td style="width: 20%;>Foto</td>
</tr>

<?php while($row = mysqli_fetch_object($ergebnis2))
{
    echo "<tr><td style=width: 20%; font-family: Arial; font-weight: bold;>$row->nr</td>";
    echo "<td style=width: 05%; font-family: Arial; font-weight: bold;>$row->abk</td>";
    echo "<td style=width: 10%; font-family: Arial; font-weight: bold;>$row->beschr</td>";
    echo "<td style=width: 05%; font-family: Arial; font-weight: bold;>$row->jahr</td>";
    echo "<td style=width: 10%; font-family: Arial; font-weight: bold;>$row->personen</td>";
    echo "<td style=width: 30%; font-family: Arial; font-weight: bold;>$row->motiv</td>";
    echo "<td style=width: 20%; font-family: Arial; font-weight: bold;>$row->minifoto</td></tr>";
}
mysqli_close($db);
?>

</tbody>
</table>

<hr>
<center>
<a target=_blank href="index.html">Übersicht</a>
</center>
<hr>
<left>
<small>&Auml;nderungsstand: 22-Jan-2019<br>
<a href="../Impressum.html">Heinz Wember</a></small><br>
</html>

```

FotoDoku fixe Spaltenbreite index.html php Rel. 7.2

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
<style type="text/css">
  @import url("FORMATE.CSS") screen;
  body { font-family:Arial,sans-serif; } body { background-color: #FFFF00; } // #FFFF00 gelb; #98FB98
hellgrün; #40E0D0 türkis; #F4A460 sandbraun;
  a:link { color:#000000; text-decoration:underline;}
  a:visited { color:#FF0000; text-decoration:underline; }
  a:focus { color:#00AA00; background-color:#FFFF77;}
  a:hover { color:#EE0000; text-decoration:none; background-color:#FFFF99; }
  a:active { color:#0000EE; background-color:#FFFF99;}
<meta name="DC.Language" content="de">
<meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
</style>
<meta name="author" content="php Rel 7.xxxx Heinz Wember Augsburg, fixe Breite, 24-Jan-2019">
</head>

<frameset rows="210px,800px">
<frameset cols="210px,800px">
  <!-- Zwei Frames, deren Inhalt hier bestimmt wird. -->
  <frame name="logo" src="logo-fix.php" scrolling="no" frameborder="no" >
  <frame name="top" src="top-fix.php" scrolling="no" frameborder="no" >
</frameset>
<frameset cols="210px,800px">
  <!-- Noch mal zwei Frames, deren Inhalt hier bestimmt wird. -->
  <frame name="navigation" src="navigation02.php" scrolling="auto" frameborder="no" >
  <frame name="content" src="fotosnachgruppeundjahrfix.php" scrolling="auto" frameborder="no">
</frameset>
</frameset>

<body>
Sie befinden sich auf der Homepage Foto-Doku2
erstellt von Heinz Wember<br>
24-Jan-2019
</body>
</noframes>
</body>
</html>
```

FotoDoku fixe Spaltenbreite php Rel. 7.2

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("FORMATE.CSS") screen;
  </style>
  <meta name="author" content="php7: Heinz Wember Augsburg, fixe Spaltenbreite, 24-Jan-2019">
  <title>Dokumentation Fotos</title>
</head>
<body>
<br>

<h1>Alle Fotos sortiert nach Gruppe und Jahr: fixe Spaltenbreite</h1>

<br>
<?php
  define("UNLOCK",1);
  include "../db/zugriff-db-fotodoku.inc.php";

  $sql = "SELECT fotos.nummer AS nr, fotos.abk AS abk, gruppe.beschreibung AS beschr, fotos.jahr AS
jahr,
        fotos.personen AS personen, fotos.motiv AS motiv, minifoto
FROM fotos
LEFT JOIN gruppe ON gruppe.abk = fotos.abk
ORDER by fotos.abk, gruppe.abk, fotos.jahr";
  $ergebnis = mysqli_query($db, $sql);
?>

<table style="width: 840px; text-align: left; margin-left: left; margin-right: auto;" border="1" cellpadding="2" cellspacing="2">
  <caption></caption><tbody>
    <tr>
      <td style="width: 60px;">Fotonummer</td>
      <td style="width: 40px;">Abk</td>
      <td style="width: 80px;">Beschreibung</td>
      <td style="width: 40px;">Jahr</td>
      <td style="width: 80px;">Personen</td>
      <td style="width:360px;">Motiv</td>
      <td style="width:180px;">Foto</td>
    </tr>
```

```

<?php
if ($ergebnis->num_rows > 0)
{
    while($row = $ergebnis->fetch_assoc())
    {
        echo "<tr><td style=width: 60px; font-family: Arial; font-weight: bold;>$row[nr]</td>";
        echo "<td style=width: 40px; font-family: Arial; font-weight: bold;>$row[abk]</td>";
        echo "<td style=width: 80px; font-family: Arial; font-weight: bold;>$row[beschr]</td>";
        echo "<td style=width: 40px; font-family: Arial; font-weight: bold;>$row[jahr]</td>";
        echo "<td style=width: 80px; font-family: Arial; font-weight: bold;>$row[personen]</td>";
        echo "<td style=width:360px; font-family: Arial; font-weight: bold;>$row[motiv]</td>";
        echo "<td style=width:180px; font-family: Arial; font-weight: bold;>$row[minifoto]</td></tr>";
    }
} else {
    echo "0 results";
}
mysqli_close($db);

?>

```

```

</tbody>
</table>

<hr>
<center>
<a target=_blank" href="index.html">Übersicht</a>
</center>
<hr>
<left>
<small>&Auml;nderungsstand: 24-Jan-2019<br>
<a href="../Impressum.html">Heinz Wember</a></small><br>
</body>
</html>

```


FotoDoku fixe Spaltenbreite php Rel. 7.2

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("FORMATE.CSS") screen;
  </style>
  <meta name="author" content="php7: Heinz Wember Augsburg, fixe Spaltenbreite, 24-Jan-2019">
  <title>Dokumentation Fotos</title>
</head>
<body>
<br>

<h1>Alle Fotos sortiert nach Gruppe und Jahr: fixe Spaltenbreite</h1>

<br>
<?php
  define("UNLOCK",1);
  include "../db/zugriff-db-fotodoku.inc.php";

  $sql = "SELECT fotos.nummer AS nr, fotos.abk AS abk, gruppe.beschreibung AS beschr, fotos.jahr AS
jahr,
        fotos.personen AS personen, fotos.motiv AS motiv, minifoto
FROM fotos
LEFT JOIN gruppe ON gruppe.abk = fotos.abk
ORDER by fotos.abk, gruppe.abk, fotos.jahr";
  $ergebnis = mysqli_query($db, $sql);
?>

<table style="width: 840px; text-align: left; margin-left: left; margin-right: auto;" border="1" cellpadding="2" cellspacing="2">
  <caption></caption><tbody>
    <tr>
      <td style="width: 60px;">Fotonummer</td>
      <td style="width: 40px;">Abk</td>
      <td style="width: 80px;">Beschreibung</td>
      <td style="width: 40px;">Jahr</td>
      <td style="width: 80px;">Personen</td>
      <td style="width:360px;">Motiv</td>
      <td style="width:180px;">Foto</td>
    </tr>
```

```

<?php
if ($ergebnis->num_rows > 0)
{
while($row = $ergebnis->fetch_assoc())
{
echo "<tr><td style=width: 60px; font-family: Arial; font-weight: bold;>$row[nr]</td>";
echo "<td style=width: 40px; font-family: Arial; font-weight: bold;>$row[abk]</td>";
echo "<td style=width: 80px; font-family: Arial; font-weight: bold;>$row[beschr]</td>";
echo "<td style=width: 40px; font-family: Arial; font-weight: bold;>$row[jahr]</td>";
echo "<td style=width: 80px; font-family: Arial; font-weight: bold;>$row[personen]</td>";
echo "<td style=width:360px; font-family: Arial; font-weight: bold;>$row[motiv]</td>";
echo "<td style=width:180px; font-family: Arial; font-weight: bold;>$row[minifoto]</td></tr>";
}
} else {
echo "0 results";
}
mysqli_close($db);

?>

```

```

</tbody>
</table>

<hr>
<center>
<a target=_blank" href="index.html">Übersicht</a>
</center>
<hr>
<left>
<small>&Auml;nderungsstand: 24-Jan-2019<br>
<a href="../Impressum.html">Heinz Wember</a></small><br>
</body>
</html>

```

FotoDoku fixe Spaltenbreite php Rel. 7.2: Suche nach Motiven

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url("FORMATE.CSS") screen;
  </style>
  <meta name="DC.Language" content="de">
  <meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
  <meta name="author" content="php Rel. 7.xxxx Heinz Wember Augsburg, fixe Spaltenbreite 24-Jan-
2019">
  <title>Fotos suche Motiv</title>
</head>
<body>
<br>
<h1>Fotos suche Motiv</h1>
<left>
<br>
<?php

  if (!$_POST['gesendet'] && !$_POST['motiv'])
  {
?>
  <form method="POST" action="<?php echo $PHP_SELF ?>">
  <p>Motiv z.B. %Zugspitze% oder %Matterhorn% <br>wenn nur % eingegeben wird, kommen alle Sätze":
  <input type="text" name="motiv" size="30">
  </p>
  <input type="submit" name="Submit" value="Submit">
  <input type="hidden" name="gesendet" value="1">
  </form>

<?php

  }

else {

  $motiv=$_POST['motiv'];
  echo "Auswahl für $motiv";
  define("UNLOCK",1);
  include "../db/zugriff-db-fotodoku.inc.php";

  $sql = "SELECT COUNT(*)
          FROM fotos
          WHERE motiv LIKE '$motiv'";

  $ergebnis = mysqli_query($db, $sql);
  $anzahl = mysqli_fetch_array($ergebnis);
  echo " : Anzahl der Sätze: $anzahl[0]<BR><BR>"; // Gibt die Anzahl aus
  mysqli_close($db);
```

```

define(„UNLOCK“,1);
include „../db/zugriff-db-fotodoku.inc.php“;
$abfrage2 = „SELECT fotos.nummer AS nr, fotos.abk AS abk, gruppe.beschreibung AS beschr, fotos.jahr
AS jahr,
    fotos.personen AS personen, fotos.motiv AS motiv, fotos.minifoto
FROM fotos
LEFT JOIN gruppe ON gruppe.abk = fotos.abk
WHERE motiv LIKE ‚$motiv‘
ORDER by fotos.abk, fotos.jahr“;

$ergebnis2 = mysqli_query($db, $abfrage2);
?>
<table style=“width: 840px; text-align: left; margin-left: left; margin-right: auto;“ border=“1“ cellpadding=“2“ cellspacing=“2“>
<caption></caption><tbody>
<tr>
<td style=“width: 60px;“>Fotonummer</td>
<td style=“width: 40px;“>Abk</td>
<td style=“width: 80px;“>Beschreibung</td>
<td style=“width: 40px;“>Jahr</td>
<td style=“width: 80px;“>Personen</td>
<td style=“width:360px;“>Motiv</td>
<td style=“width:180px;“>Foto</td>
</tr>

<?php while($row = $ergebnis2->fetch_assoc())
{
    echo „<tr><td style=width: 60px; font-family: Arial; font-weight: bold;>$row[nr]</td>“;
    echo „<td style=width: 40px; font-family: Arial; font-weight: bold;>$row[abk]</td>“;
    echo „<td style=width: 80px; font-family: Arial; font-weight: bold;>$row[beschr]</td>“;
    echo „<td style=width: 40px; font-family: Arial; font-weight: bold;>$row[jahr]</td>“;
    echo „<td style=width: 80px; font-family: Arial; font-weight: bold;>$row[personen]</td>“;
    echo „<td style=width:360px; font-family: Arial; font-weight: bold;>$row[motiv]</td>“;
    echo „<td style=width:180px; font-family: Arial; font-weight: bold;>$row[minifoto]</td></tr>“;
}
mysqli_close($db);
}
?>

</tbody>
</table>

<hr>
<center>
<a target=_blank“ href=“index.html“>Übersicht</a>
</center>
<hr>
<left>
<small>&Auml;nderungsstand: 12-Jul-2019<br>
<a href=“../Impressum.html“>Heinz Wember</a></small><br>
</html>

```

AusgabeAuswahl nach Motiv-fix

Das Bearbeitet Ansicht Ansicht Leuchten Datei Hilfe

heinz-wember.de/MySQL-fix-Foto/ X Dokumentation Heinz Wember X +

https://heinz-wember.de/MySQL-fix-Foto/Doku/index.html

Foto-Doku: fixe Spaltenbreite erstellt von Heinz Wember 22-Jan-2019

Dokumentation der Fotos: fixe Spaltenbreite

Fotonummer/Adi	Beschreibung	Jahr	Personen	Motiv	Foto
----------------	--------------	------	----------	-------	------

FOTOS SUCHEN NACH MOTIV

Auswahl für %Hochgrat% Anzahl der Sätze: 1

Fotonummer/Adi	Beschreibung	Jahr	Personen	Motiv	Foto
894614 (79) jpg	berg	Bergfotos	2009	Gipfel Hochgrat nach SW, links Geldkern und Zugspitze, Mitte Hochgrat, rechts Tretsch, Mafelkogel und Hochtrappitze	

[Übersicht](#)

Änderungsdatum: 12 Jul 2015
[Heinz Wember](#)

Navigation
fixe Spaltenbreite
Übersicht Dokumentation

MySQL Version anzeigen
PHP Version anzeigen

Den... nach Gruppe und Jahr
Fotos nach Gruppe und Jahr
Auswahl Person
Auswahl Motiv
Abkürzungen

Version relativ Spaltenbreite

Datenbank Konsole und
Erweiterungen
weitere Dokumentation eines
ähnlichen Systems
Dokumentation SP1P und
Teamviewer
Änderungsdatum: 24-Jan-2019

Zur Suche Text hier eingeben

9:19
21.01.2025

Verwendung von html-Steuerzeichen in Tabellen

Dokumentation: Pflege der Datenbanktabellen

Häufig vorkommende Steuerzeichen:

`
` neue Zeile

`... ` Zeichen fett in Farbe

` ... ` Zeichen fett in schwarz

`<big> </big>` eine Zeichengröße größer

`<small </small>` eine Zeichengröße kleiner

`` Aufrufen (direkt Einbinden) einer Abbildung

`verschiedene Funktionen von phpmyadmin` Link: dieser Link ruft das Programm `dokuupdatemysql1.php` auf und zeigt es in demselben Fenster, von dem das Programm aufgerufen wurde

`verschiedene Funktionen von phpmyadmin` Link: dieser Link ruft das Programm `dokupdb1.php` auf und zeigt es in dem Fenster `content`

Änderungsstand: 04-Mar-2019

Heinz Wember

Besonderheiten mit Frameset

Besonderheiten mit FRAMESET

1. Die Suchmaschinen (z.B. Google) adressieren nicht nur die Seiten, bei denen im Coding die Frameset-Definitionen enthalten sind (z.B. bei mir heinz-wember.de/My-SQL-Kriegshaber/index.html oder aber die Adresse heinz-wember.de/My-SQL-Kriegshaber/index-kat.html), sondern auch diejenigen Adressen, die Teil eines Framesets sind, z.B. bei heinz-wember.de/My-SQL-Kriegshaber/hausnummern.html.

In diesem Fall wird also nur dieses eine Fenster (bei mir content genannt) angezeigt, der Zusammenhang mit dem Fenster navigation und auch top wird also nicht angezeigt. Man kann also nicht erkennen, von wo diese Seite aufgerufen wurde (in meinem Fall war dies heinz-wember.de/My-SQL-Kriegshaber/index.html mit dem Aufruf aus dem Fenster navigation die Adresse von [Hausnummern.html](http://heinz-wember.de/My-SQL-Kriegshaber/hausnummern.html).) Deshalb habe ich mir angewöhnt, unten auf der genannten Seite einen Link zu platzieren, der den definierten Frameset enthält, in diesem Fall also die Adresse heinz-wember.de/My-SQL-Kriegshaber/index.html mit dem Linktext Straßen und Hausnummern. Als Inhalt dieses Links ist nicht nur der Wert href sondern auch der Wert target anzuwenden. Siehe unter 2.

2. Die Programmierung eines Rücksprungslinks

Zur Programmierung der Seiten verwende ich entweder vi (bzw. winvii) oder das relativ alte Programm NVU. Hat dieses Unterprogramm, wie oben im Beispiel angegeben, die Endung html, so kann man NVU benutzen.

Die verschiedenen Werte des Parameters target.

2.1. Wert target fehlt bzw. hat den Wert ""

Bei NVU wird geöffnet in diesem Frame, Wert target "" bzw der Parameter target fehlt
Rücksprung zur Startseite von MySQL-Kriegshaber mit Wert target="" und href="../index.html"

2.2. Wert target hat den Wert "_parent"

bei NVU Ziel wird geöffnet im übergeordneten Frameset (Wert target _top)
Rücksprung zur Startseite von MySQL-Kriegshaber mir Wert target="_parent" und href="../index.html"

2.3 Wert target hat den Wert "_self"

bei NVU Ziel wird geöffnet mit aktuellen Framen the current Frame
Rücksprung zur Startseite von MySQL-Kriegshaber mit Wert target="_self" und href="../index.html"

2.4 Wert target hat den Wert "_blank"

bei NVU Ziel wird geöffnet im neuen Fenster
Rücksprung zur Startseite von MySQL-Kriegshaber mit Wert target="_blank" und href="../index.html"

2.5. Wert target hat den Wert "_top"

bei NVU Ziel wird geöffnet im gleichen Frame (ohne Frameset)
Rücksprung zur Startseite von MySQL-Kriegshaber mit Wert target="_top" und href="../index.html"

Bei meiner Applikation MySQL-Kriegshaber verwende ich nur die Möglichkeit 2.1, hier soll die aufgerufene Seite im Fenster content (das große rechte Fenster) aufgerufen werden.

Sinnvoll ist dies z.B. bei Aufruf der Seite weitereSeite siehe den Link hierzu weiter unten.

Dann ist bei Aufruf der Startseite die Möglichkeit 2.4 sinnvoll.

Bei Aufruf eines Unterprogrammes der Navigation z.B.[index-kat.html](http://heinz-wember.de/My-SQL-Kriegshaber/index-kat.html) nehme ich ebenfalls die Version 2.4.

Die beiden Programme index.html und index-kat.html als txt-Datei

```
<!DOCTYPE html PUBLIC „-//W3C//DTD XHTML 1.0 Transitional//EN“ „http://www.w3.org/TR/xhtml1/
DTD/xhtml1-transitional.dtd“>
<html><head>
<style type=“text/css“>
  @import url(„FORMATE.CSS“) screen;
</style>
<title>Straßen, Häuser und Kataster von Kriegshaber</title>
<meta name=“author“ content=“Heinz Wember Augsburg“>
<meta http-equiv=“Content-Type“ content=“text/html; charset=ISO-8859-1“ />
</head>
<frameset rows=“130,*“>
<!-- Dadurch ergeben sich zwei Frame-Fenster, deren Inhalt hier bestimmt wird.
  Das erste Frame-Fenster wird normal definiert, anstelle des zweiten
  folgt jedoch das folgende untergeordnete Frameset: -->
<frame name=“top“ src=“top.php“ scrolling=“no“ frameborder=“no“>
<frameset cols=“15%,85%“rows=1200 >
<frame name=“navigation“ src=“navigation.php“ scrolling=“auto“ frameborder=“no“ >
<frame name=“content“ src=“strassennamen.php „, scrolling=“auto“ frameborder=“no“>
</frameset>
</frameset>
<body>
Datei index.html
</body></noframes></body></html>
```

```
<!DOCTYPE html PUBLIC „-//W3C//DTD HTML 4.01 Transitional//EN“>
<html xmlns=“http://www.w3.org/1999/xhtml“ xml:lang=“de“>
<head>
<style type=“text/css“>
  @import url(„FORMATE.CSS“) screen;
  body { font-family:Arial,sans-serif; }
  a:link { color:#000000; text-decoration:underline;}
  a:visited { color:#FF0000; text-decoration:underline; }
  a:focus { color:#00AA00; background-color:#80FF80;}
  a:hover { color:#EE0000; text-decoration:none; background-color:#FFFF99; }
  a:active { color:#0000EE; background-color:#FFFF99;}
</style>
<meta name=“author“ content=“Heinz Wember Augsburg, php7, 18-Feb-2019“>
</head>
<frameset rows=“130,*“>
<!-- Dadurch ergeben sich zwei Frame-Fenster, deren Inhalt hier bestimmt wird.
  Das erste Frame-Fenster wird normal definiert, anstelle des zweiten
  folgt jedoch das folgende untergeordnete Frameset: -->
<frame name=“top“ src=“top.php“ scrolling=“no“ frameborder=“no“>
<frameset cols=“15%,85%“ rows=1200>
<frame name=“navigation“ src=“untermenu30.php“ scrolling=“auto“ frameborder=“no“ >
<frame name=“content“ src=“dokukat1840system.php“ scrolling=“auto“ frameborder=“no“>
</frameset>
</frameset>
<body>
Datei index-kat.html<br>
</body></noframes></body></html>
```


Vergleich der beiden Programme navigation.php und untermenu30.php

Es wird nur der Befehl gezeigt, der unterschiedlich ist:

bei Programm navigation.php:

```
$abfrage = "SELECT gruppe , ugruppe , titel
            FROM navigation
            WHERE gruppe = '00' AND aktiv = 'j'
            ORDER BY ugruppe";
```

bei Programm untermenu30.php:

```
$abfrage = "SELECT gruppe , ugruppe, aktiv, titel
            FROM navigation
            WHERE gruppe = '30' AND aktiv = 'j'
            ORDER BY ugruppe";
```

Hier zwei Beispiele für den Aufruf eines Programms in der Tabelle navigation:

Beispiel für den ersten Aufruf:

```
<a target="content" href="dokukat1840system.php">Systematik des Katasters</a>
```

Beispiel für die weitereSeite

```
<a target="content" href="dokukat1840system.php">Systematik des Katasters</a>
```

Hier bleiben die Fenster navigation und top unverändert.

Beispiel für eine weitereSeite, wobei hier das Navigationsfenster neu erstellt wird

```
<a target="navigation" href="untermenu32.php">Documentation in english</a>
```

Der Vorteil dieser Vorgehensweise ist, dass man gleichzeitig ein anderes Startprogramm für das große Fenster (frame content) anwählen kann, ansonsten sind die beiden Aufrufprogramme gleich.

Der Unterschied von Programm navigation.php zu untermenu30.php ist minimal siehe oben auf dieser Seite

Wird bei Aufruf eines Programms des Typs index.html, das die Frames und die dazugehörigen Aufrufprogramme enthält, das Programm für den Frame content nicht gefunden, bleibt, bei Aufruf das Fenster content leer (es kommt auch keine Fehlermeldung).

3. Test der verschiedenen Möglichkeiten des Werten für target.

3.1. Zuerst Aufruf von index.html, damit sind die Frames definiert.

3.2. Nun Aufruf dieses Dokumentationsprogrammes mit target="content".

3.3. Nun können die obenstehenden Links unter 2. nacheinander aufgerufen werden, die Wirkung des verschiedenen Werte von target zu demonstrieren.

Heinz Wember
25-Mar-2020

The following example is documented in english

Example database for music titels structure of the database

```
-- phpMyAdmin SQL Dump
-- version 4.6.6deb4
-- https://www.phpmyadmin.net/
--
-- Host: infongd-eu6168.rtr.schlund.de
-- Erstellungszeit: 07. Okt 2019 um 07:14
-- Server-Version: 5.5.60-0+deb7u1
-- PHP-Version: 7.0.33-0+deb9u5

SET SQL_MODE = „NO_AUTO_VALUE_ON_ZERO“;
SET time_zone = „+00:00“;

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Datenbank: `db336777968`
--
CREATE DATABASE IF NOT EXISTS `db336777968` DEFAULT CHARACTER SET latin1 COLLATE latin1_swedish_ci;
USE `db336777968`;

-----

--
-- Tabellenstruktur für Tabelle `dokumentation`
--
CREATE TABLE `dokumentation` (
  `id` int(4) NOT NULL,
  `text` text COLLATE latin1_german1_ci NOT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Musiktitel Dokumentation';

-----

CREATE TABLE `documentation` (
  `id` int(4) NOT NULL,
  `text` text COLLATE latin1_german1_ci NOT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='titel of music documentation';

-----
```

```
--
-- Tabellenstruktur für Tabelle `dokuradiosendungen`
--

CREATE TABLE `dokuradiosendungen` (
  `id` int(4) NOT NULL,
  `text` text COLLATE latin1_german1_ci NOT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Musiktitel
Dokumentation';
```

```
--
-- Tabellenstruktur für Tabelle `musiktitel`
--
```

```
CREATE TABLE `musiktitel` (
  `id` int(4) NOT NULL,
  `titelname` text COLLATE latin1_german1_ci,
  `gruppe` varchar(10) COLLATE latin1_german1_ci DEFAULT 'Classical',
  `interpret` text COLLATE latin1_german1_ci,
  `album` text COLLATE latin1_german1_ci,
  `groesse` varchar(20) COLLATE latin1_german1_ci DEFAULT NULL,
  `dauer` varchar(20) COLLATE latin1_german1_ci DEFAULT NULL,
  `jahr` year(4) DEFAULT NULL,
  `drucken` varchar(2) COLLATE latin1_german1_ci DEFAULT NULL,
  `erstellungsdatum` varchar(20) COLLATE latin1_german1_ci DEFAULT '04.11.2013',
  `updatedatum` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Musiktitel mp3';
```

```
--
-- Tabellenstruktur für Tabelle `navigation`
--
```

```
CREATE TABLE `navigation` (
  `id` tinyint(4) NOT NULL,
  `gruppe` char(2) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL DEFAULT '00',
  `ugruppe` char(2) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL,

  `aktiv` varchar(1) CHARACTER SET latin1 COLLATE latin1_german1_ci NOT NULL DEFAULT 'j',
  `titel` varchar(255) CHARACTER SET latin1 COLLATE latin1_german1_ci DEFAULT NULL,
  `text` varchar(255) CHARACTER SET latin1 COLLATE latin1_german1_ci DEFAULT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

```
--
-- Tabellenstruktur für Tabelle `rtsdokumentation`
--
```

```
CREATE TABLE `rtsdokumentation` (
```

```

`id` int(4) NOT NULL,
`text` text COLLATE latin1_german1_ci NOT NULL,
`erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Musiktitel Dokumentation';

```

```

--
-- Tabellenstruktur für Tabelle `titel`
--

```

```

CREATE TABLE `titel` (
  `id` int(10) NOT NULL,
  `titelname` text COLLATE latin1_german1_ci,
  `interpret` text COLLATE latin1_german1_ci,
  `komponist` text COLLATE latin1_german1_ci,
  `album` text COLLATE latin1_german1_ci,
  `genre` text COLLATE latin1_german1_ci,
  `groesse` int(10) DEFAULT NULL,
  `dauer` int(10) DEFAULT NULL,
  `jahr` year(4) DEFAULT NULL,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Musikttitel mp3';

```

```

--
-- Tabellenstruktur für Tabelle `titel20101124`
--

```

```

CREATE TABLE `titel20101124` (
  `titelname` text COLLATE latin1_german1_ci,
  `interpret` text COLLATE latin1_german1_ci,
  `album` text COLLATE latin1_german1_ci,
  `groesse` int(10) DEFAULT NULL,
  `dauer` int(10) DEFAULT NULL,
  `jahr` year(4) DEFAULT NULL,
  `ort` text COLLATE latin1_german1_ci,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Musikttitel mp3';

```

```

--
-- Tabellenstruktur für Tabelle `titelmozart`
--

```

```

CREATE TABLE `titelmozart` (
  `titelname` text COLLATE latin1_german1_ci,
  `interpret` text COLLATE latin1_german1_ci,
  `album` text COLLATE latin1_german1_ci,
  `groesse` int(10) DEFAULT NULL,
  `dauer` int(10) DEFAULT NULL,
  `jahr` year(4) DEFAULT NULL,
  `ort` text COLLATE latin1_german1_ci,
  `erstellt` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP
) ENGINE=MyISAM DEFAULT CHARSET=latin1 COLLATE=latin1_german1_ci COMMENT='Musikttitel mp3';

```

```

--
-- Indizes der exportierten Tabellen
--
--
-- Indizes für die Tabelle `dokumentation`
--
ALTER TABLE `dokumentation`
  ADD PRIMARY KEY (`id`);

--
-- Indizes für die Tabelle `dokuradiosendungen`
--
ALTER TABLE `dokuradiosendungen`
  ADD PRIMARY KEY (`id`);

--
-- Indizes für die Tabelle `musiktitel`
--
ALTER TABLE `musiktitel`
  ADD PRIMARY KEY (`id`),
  ADD KEY `titelname` (`titelname`(10));

--
-- Indizes für die Tabelle `navigation`
--
ALTER TABLE `navigation`
  ADD PRIMARY KEY (`id`),
  ADD KEY `gruppe` (`gruppe`);

--
-- Indizes für die Tabelle `rtsdokumentation`
--
ALTER TABLE `rtsdokumentation`
  ADD PRIMARY KEY (`id`);

--
-- Indizes für die Tabelle `titel`
--
ALTER TABLE `titel`
  ADD PRIMARY KEY (`id`),
  ADD KEY `jahr` (`jahr`),
  ADD KEY `grosesse` (`grosesse`);

--
-- Indizes für die Tabelle `titel20101124`
--
ALTER TABLE `titel20101124`
  ADD KEY `titel` (`titelname`(20));

--
-- Indizes für die Tabelle `titelmozart`
--
ALTER TABLE `titelmozart`
  ADD KEY `titel` (`titelname`(20));

--

```

```

-- AUTO_INCREMENT für exportierte Tabellen
--
--
-- AUTO_INCREMENT für Tabelle `dokumentation`
--
ALTER TABLE `dokumentation`
  MODIFY `id` int(4) NOT NULL AUTO_INCREMENT;
--
-- AUTO_INCREMENT für Tabelle `dokuradiosendungen`
--
ALTER TABLE `dokuradiosendungen`
  MODIFY `id` int(4) NOT NULL AUTO_INCREMENT;
--
-- AUTO_INCREMENT für Tabelle `musiktitel`
--
ALTER TABLE `musiktitel`
  MODIFY `id` int(4) NOT NULL AUTO_INCREMENT;
--
-- AUTO_INCREMENT für Tabelle `navigation`
--
ALTER TABLE `navigation`
  MODIFY `id` tinyint(4) NOT NULL AUTO_INCREMENT;
--
-- AUTO_INCREMENT für Tabelle `rtsdokumentation`
--
ALTER TABLE `rtsdokumentation`
  MODIFY `id` int(4) NOT NULL AUTO_INCREMENT;
--
-- AUTO_INCREMENT für Tabelle `titel`
--
ALTER TABLE `titel`
  MODIFY `id` int(10) NOT NULL AUTO_INCREMENT;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

```

Documentation of two programs in english

programs:

e-documentation.php

```
<!DOCTYPE HTML PUBLIC „-//W3C//DTD HTML 4.0 Transitional//EN“>
<html xmlns=“http://www.w3.org/1999/xhtml“ xml:lang=“de“>
<head>
  <style type=“text/css“>
    @import url(„FORMATE.CSS“) screen;
  </style>
  <title>Description who to enter the data</title>
  <meta name=“author“ content=“Heinz Wember Augsburg, php7, 07-Oct-2019“>
</head>
<br>
<h1>Musictitel - Documentation</h1>

<body>
<br>
<br>
<br>
<table style=“text-align: left; margin-left: left; margin-right: auto; border=1 cellpadding=2 cellspacing=2>

  <tbody>

<?php
define(„UNLOCK“,1);
include „../db/zugriff-dbmusiktitel.inc.php“;

$abfrage = „SELECT text from documentation“;

$ergebnis = mysqli_query($db, $abfrage);
while($row = $ergebnis->fetch_assoc())
{
  echo „<tr><td style=width: 10%; font-family: Arial; font-weight: bold;> $row[text]</td></tr>“;
}
mysqli_close($db);
?>

</tbody>
</table>
<hr>
<center>
<a target=“_blank“ href=“index.html“>Scope</a>
<hr>
</center>
<left>
<small>updated: 07-Oct-2019<br>
<a href=“../Impressum-englisch.html“>Heinz Wember</a></small>
</body>
</html>
```

program selection.php

```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="de">
<head>
  <style type="text/css">
    @import url(„FORMATE.CSS“) screen;
  </style>
  <meta name="author" content="Heinz Wember Augsburg, php7, 10-Oct-2019">
  <title>titels for music: selection </title>
</head>
<body>
<br>

<h1>titels for music: selection </h1>

<left>
<br>

<?php
include „../db/zugriff-dbmusiktitel.inc.php“;

if (!$_POST[„gesendet“] && !$_POST[„text“])
{
?>

<form method="POST" action="<?php echo $PHP_SELF ?>">
<p>composer e.g. %Bach% or %Mozart%<br>
titel of music like %Sinfonie% or %Konzert%<br>
beginning of a song like %Te%Deum% or %Exultate%Jubilate%<br>
artist e.g. %Helmuth Rilling% or %Kirkby% or %London% or %1052% at BWV or %315% at KV;<br>
you can also search for instruments such as %Klavier% or %Violine%<br>
it is essential that the search will be done in the field Komponist (composer) and Werk (titel of music) but also in the
field Interpreten (actors)<br>
if you search for %Bach% then you will get any records containing the word Bach but also e.g. Bach-
Collegium<br>
if you search with the entry Bach% then the word Bach have to be located at the beginning of the field<br>
if you search for Bach-Collegium then you have to enter %Bach-Collegium%<br>

  <input type="text" name="text" size="30">
</p>

  <input type="submit" name="Submit" value="Submit">
  <input type="hidden" name="gesendet" value="1">

</form>

<?php
}

else {

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



```

$abfrage = „SELECT titelname, interpret, album, groesse, dauer, jahr, erstellungsdatum
FROM musiktitel
WHERE interpret LIKE ‚$text‘
OR titelname LIKE ‚$text‘
OR album LIKE ‚$text‘
ORDER by titelname“;
$ergebnis = mysqli_query($db, $abfrage);

```

```

echo „Selection for $text“;

```

```

$result=mysqli_query($db, „SELECT COUNT(*)
FROM musiktitel
WHERE interpret LIKE ‚$text‘
OR titelname LIKE ‚$text‘
OR album LIKE ‚$text‘“);

```

```

$anzahl = mysqli_fetch_array($result);
echo „ : number of records: $anzahl[0]<BR><BR>“; // Gibt die Anzahl aus
?>

```

```

<table style=“width: 85%; text-align: left; margin-left: left; margin-right: auto;“ border=“1“ cellpadding=“2“ cellspacing=“2“>

```

```

<caption></caption><tbody>
<tr>
<td style=“width: 10%;“>Komponist/Werk<br></td>
<td style=“width: 40%;“>Interpreten</td>
<td style=“width: 15%;“>Album </td>
<td style=“width: 3%;“>Größe MB </td>
<td style=“width: 3%;“>Dauer min </td>
<td style=“width: 4%;“>Jahr </td>
<td style=“width: 10%;“>Ort </td>
</tr>

```

```

<?php
while($row = $ergebnis->fetch_assoc())
{
echo „<tr><td style=width: 10%; font-family: Arial; font-weight: bold;>$row[titelname]</td>“;
echo „<td style=width: 40%; font-family: Arial; font-weight: bold;>$row[interpret]</td>“;
echo „<td style=width: 15%; font-family: Arial; font-weight: bold;>$row[album]</td>“;
echo „<td style=width: 3%; font-family: Arial; font-weight: bold;>$row[groesse]</td>“;
echo „<td style=width: 3%; font-family: Arial; font-weight: bold;>$row[dauer]</td>“;
echo „<td style=width: 4%; font-family: Arial; font-weight: bold;>$row[jahr]</td>“;
echo „<td style=width: 10%; font-family: Arial; font-weight: bold;>$row[erstellungsdatum]</td></tr>“;
}
mysqli_close($db);
?>

```

```

</tbody>
</table>

```

```

<?php
}
?>
<br>

```

The search for fields allow multiple entries, e.g. %Fritz%Wunderlich% where the search is successful if both fields are entered into the database in this order. The search %Wunderlich%Fritz% is not successful for all entries which was entered into the database like Fritz Wunderlich

These are conj selections, or selections are not possible with this program. You have to do two searches

<hr>

<center>

Scope

<hr>

</center>

<left>

<small>updated 10-Oct-2019

Heinz Wember</small>

</body>

</html>

Output from the function e-documentation.php

Description who to enter the data

Musicitel - Documentation

The basic for entering the data are documented here [hier](#)

field titelname composer and musical work. There are different categories:

G for religious work

K for concerts

Ka for chamber music

L for songs

M for mass

O for orchestral work

Op for opera

S for sinfonies

P for piano

St for strings

aso.

followed by the name of work, registration number such as Koechel or opus directory of works of Bach, also followed by an additional field for sorting, to make sure, the whole entry is unique followed by additional descriptions for this specific work

for field interpret: all artists should be entered

several artists are normal, the function search can seek for every name, it is possible to find every artist independently at which position of the field the name is entered

the data (composer_work, interpret, album, length of music, data size) will be entered by a csv file, therefore the ; may not be part of this field

updated: 25-Mar-2020

[Scope](#)

Updated e-documentation.php: 19-Feb-2020

[Heinz Wember](#)

Output from function selection-hochformat.php

titels for music: selection

titels for music: selection

Selection for %Kirkby% : number of records: 7

Komponist/Werk	Interpreten	Album	Größe MB	Dauer min	Jahr	Erstellungsdatum
Mozart_G_108_Regina_coeli	KirkbyEmma , Westminster Cathedral Boys Choir, Academy of Ancient Music, Ltg: Christopher Hogwood	Mozart Exultate Jubilate, Motetten	17,7 MB	13:35:00	1983	21.06.2011 13:37
Mozart_G_127_Regina_coeli	Kirkby Emma , Westminster Cathedral Boys Choir, Academy of Ancient Music, Ltg: Christopher Hogwood	Mozart Exultate Jubilate, Motetten	19,3 MB	14:41:00	1983	17.11.2008 12:17
Mozart_G_143_02_Ergo_interest	Kirkby Emma , Orchestra of the Academy of Ancient Music, Ltg: Christopher Hogwood	Mozart Exultate Jubilate, Motetten	6,6 MB	05:20:00	1983	16.08.2010 11:05
Mozart_G_165_01_Exultate_Jubilate_Exultate	Kirkby Emma , Orchestra of the Academy of Ancient Music, Ltg: Christopher Hogwood	Mozart Exultate Jubilate, Motetten	5,6 MB	04:36:00	1983	21.06.2011 13:37
Mozart_G_165_02_Exultate_Jubilate_Tandem_advenit_hora	Kirkby Emma , Orchestra of the Academy of Ancient Music, Ltg: Christopher Hogwood	Mozart Exultate Jubilate, Motetten	950,2 KB	00:47:00	1983	16.08.2010 11:03

file:///F:/Wember/Heinz/BuchMySQL/Auswertungen/tmp/Kirby.html[26.03.2020 04:44:01]

titels for music: selection

Mozart_G_165_03_Exultate_Jubilate_Tu_virginum_corona	Kirkby Emma , Orchestra of the Academy of Ancient Music, Ltg: Christopher Hogwood	Mozart Exultate Jubilate, Motetten	7 MB	05:36:00	1983	21.06.2011 13:37
Mozart_G_165_04_Exultate_Jubilate_Alleluja	Kirkby Emma , Orchestra of the Academy of Ancient Music, Ltg: Christopher Hogwood	Mozart Exultate Jubilate, Motetten	3,2 MB	02:34:00	1983	21.06.2011 13:37

The search for fields allow multiple entries, e.g. %Fritz%Wunderlich% where the search is successful if both fields are entered into the database in this order. The search %Wunderlich%Fritz% is not successful for all entries which was entered into the database like Fritz Wunderlich
These are conj selections, or selections are not possible with this program. You have to do two searches

[Scope](#)

Update: selection-hochformat.php: 25-Mar-2020
[Heinz Wember](#)

Über den Autor:

1941 geboren in Hausham/Obb

1947-51 Volksschule in Hausham

1951/60 Gymnasium in Miesbach

1960/61 Praktikum im Elektrizitätswerk in Tegernsee

1961/62 Gebirgsjäger in Mittenwald

1962/68 Studium der Elektrotechnik an der TU München

1968/96 Systemanalytiker und EDV-Leiter bei NCR in Augsburg

seit 1999 Gasthörer der Geschichte an der Uni Augsburg



