

Script generated by TTT

Title: Einf_HF (07.05.2012)

Date: Mon May 07 14:17:23 CEST 2012

Duration: 92:18 min

Pages: 51



Entity-Relationship-Diagramm: logisches Modell einer Datenbank. Für Implementierung in DBS: physikalisches Modell nötig.

Beispiel: relationales Datenbankmodell.

Relationales Modell

Tabellendarstellung

Normalisierung

Umsetzung des ER-Modells

Sichten

Verwendung von Sichten zur Auswahl einer Teilmenge der Tabellendaten.

Beispiel für eine Sicht: Vorname und Nachname von Kunden, die in München wohnen.

Views sind damit nichts anderes als benannte Such-Abfragen (z.B. SQL-Anfragen in MS ACCESS mit Namen).

Abfragesprache SQL

Beispielsysteme

mySQL (Open Source), Microsoft Access, Microsoft SQL Server, Oracle, DB2 (IBM), Sybase, Informix

Microsoft Access

Generated by Targeseam



MS ACCESS stellt zum Erstellen von Tabellen, Abfragen usw. im Allgemeinen Assistenten zur Verfügung. Trotzdem ist auch das direkte Arbeiten mit SQL möglich!

Ansichten in MS Access

MS ACCESS unterscheidet bei Tabellen und Anfragen grundsätzlich:

die **Datenblattansicht** : Anzeige der Instanz einer Basistabelle bzw. der Ergebnistabelle einer Anfrage; erlaubt auch die Eingabe, die Änderung und das Löschen von Daten in einer Basistabelle.

die **Entwurfsansicht** : interaktives Erzeugen von Tabellen bzw. Anfragen "ohne" SQL.

die **SQL-Ansicht** : direktes Arbeiten mit SQL.

SQL-Ansicht

Generated by Targeseam



Die SQL-Ansicht erlaubt

die Definition von Tabellen (CREATE TABLE),

das Ändern von Tabellen (ALTER TABLE),

das Löschen von Tabellen (DROP TABLE),

das Einfügen von Daten in Tabellen (INSERT INTO),

das Ändern von Daten in Tabellen (UPDATE),

das Löschen von Daten in Tabellen (DELETE) und

die Erstellung von SQL-Anfragen (SELECT).

Generated by Targeseam

ale Datenbanksysteme - Windows Internet Explorer

C:\www\inf-ss12\flash\inf_course3.4.html

Relationale Datenbanksysteme

Relationale Datenbanksysteme

Entity-Relationship-Diagramm: logisches Modell einer Datenbank. Für Implementierung in DBS: physikalisches Modell nötig.

Beispiel: relationales Datenbankmodell.

[Relationales Modell](#)

[Tabellendarstellung](#)

[Normalisierung](#)

[Umsetzung des ER-Modells](#)

Sichten

Verwendung von Sichten zur Auswahl einer Teilmenge der Tabellendaten.

Beispiel für eine Sicht: Vorname und Nachname von Kunden, die in München wohnen.

Views sind damit nichts anderes als benannte Such-Abfragen (z.B. SQL-Anfragen in MS ACCESS mit Namen).

[Abfragesprache SQL](#)

Beispielsysteme

Start | Relationale Datenban... | Notiz1 - Windows-Jou... | Microsoft Access | 14:22

Windows-Journal

Notiztitel

07.05.2012

1 / 1

Start | Relationale Datenban... | Notiz1 - Windows-Jou... | Microsoft Access | 14:22

Microsoft Access

Frage hier eingeben

Neue Datei

- Neu
 - Leere Datenbank...
 - Leere Datenzugriffsseite...
 - Projekt aus bestehenden Daten...
 - Projekt aus neuen Daten...
 - Aus bestehender Datei...
- Vorlagen
 - Onlinesuche nach:
 - Vorlagen auf Office Online
 - Auf meinem Computer...

Bereit

Start | Relationale Datenban... | Notiz1 - Windows-Jou... | Microsoft Access | 14:24

Microsoft Access

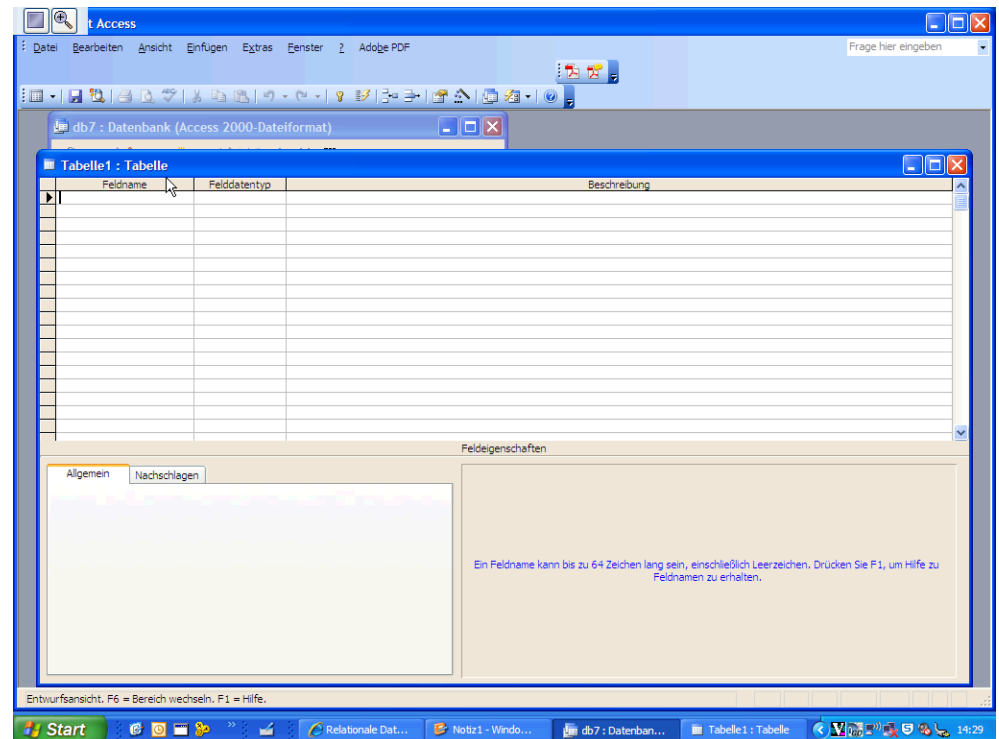
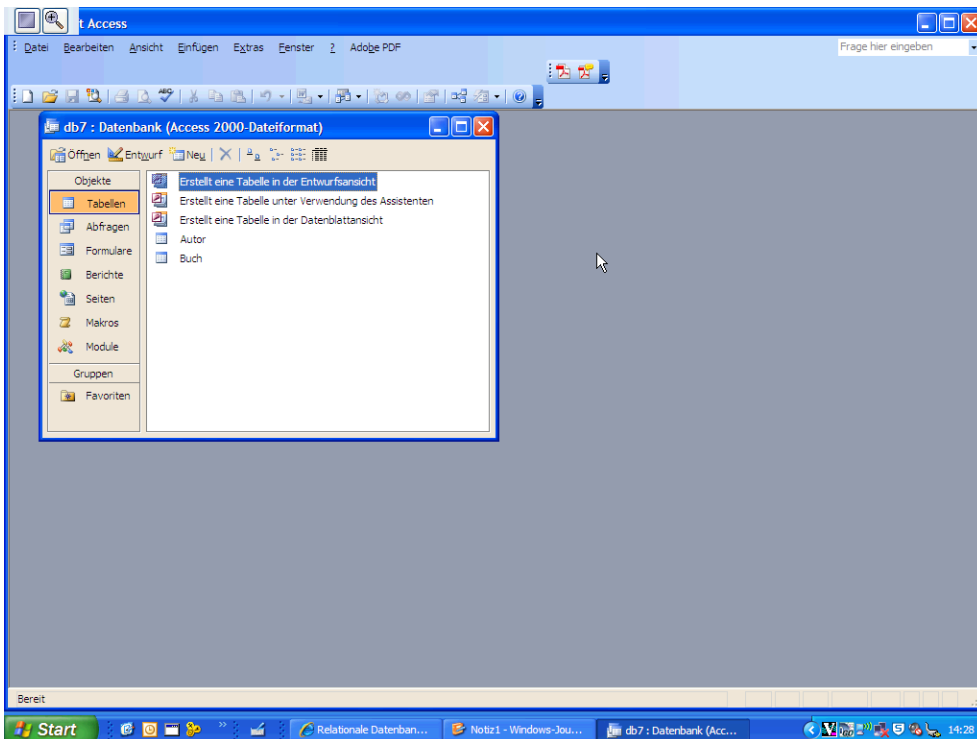
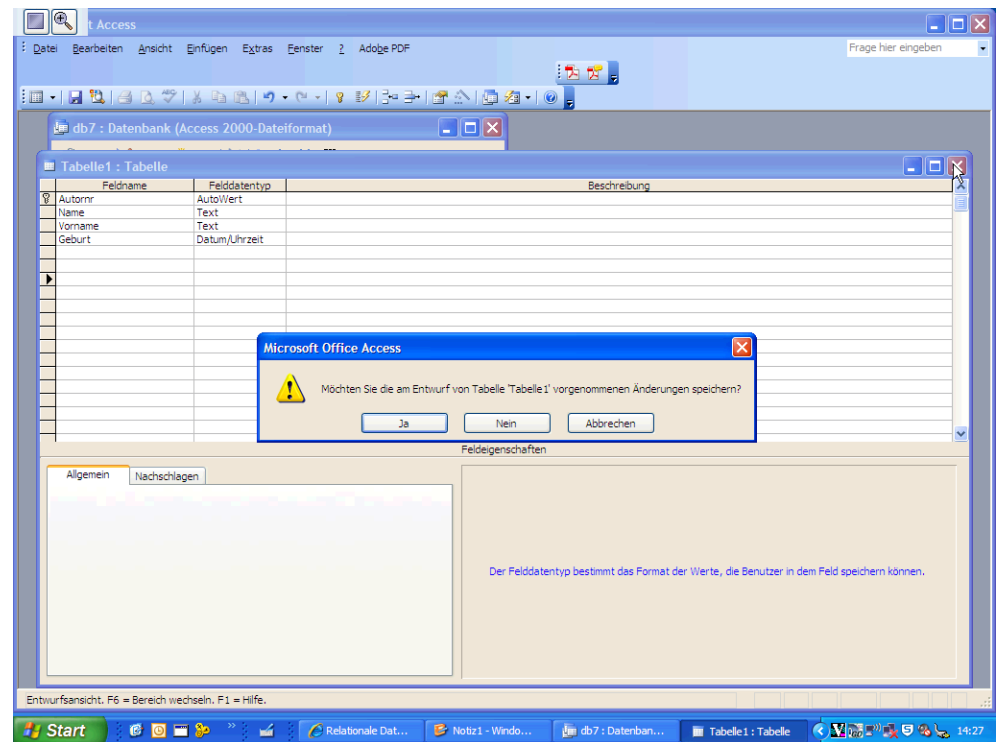
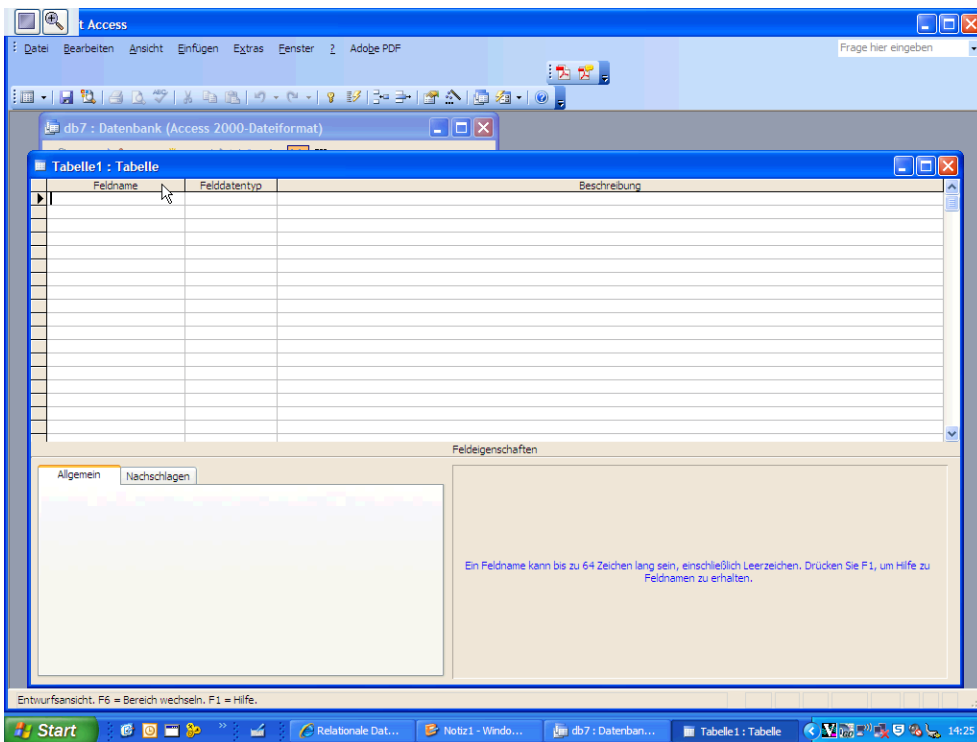
Frage hier eingeben

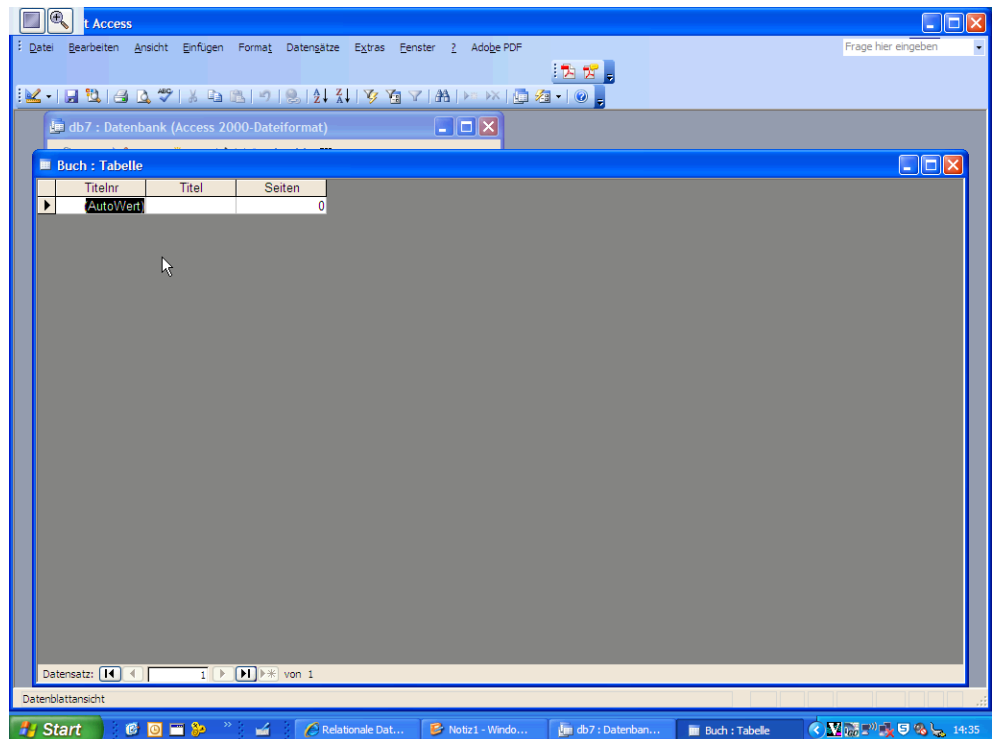
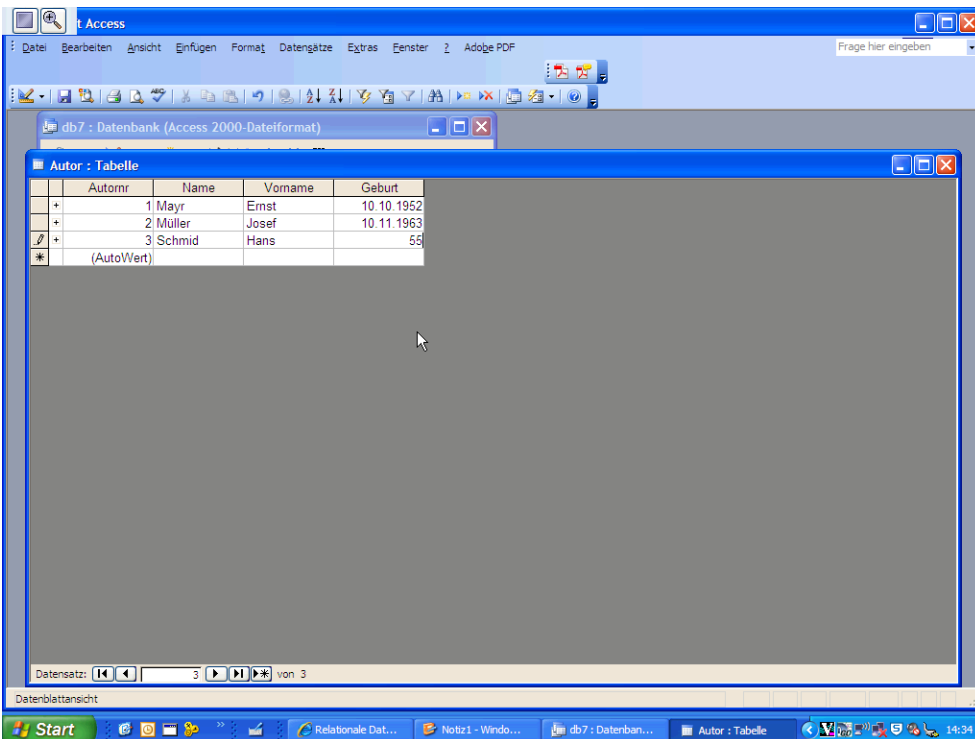
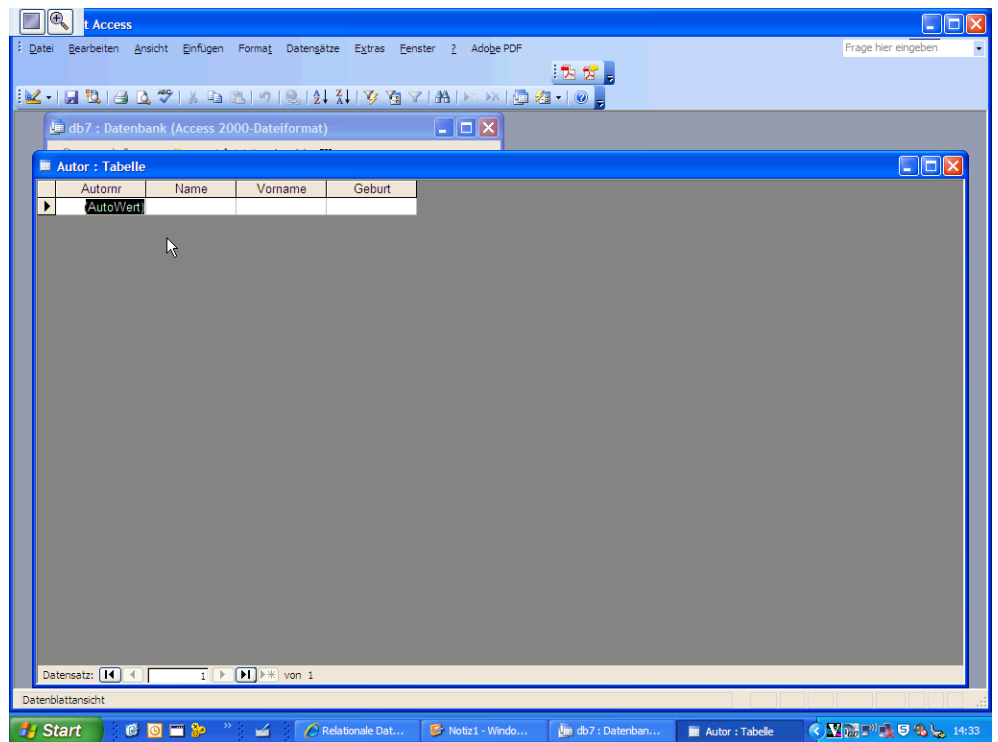
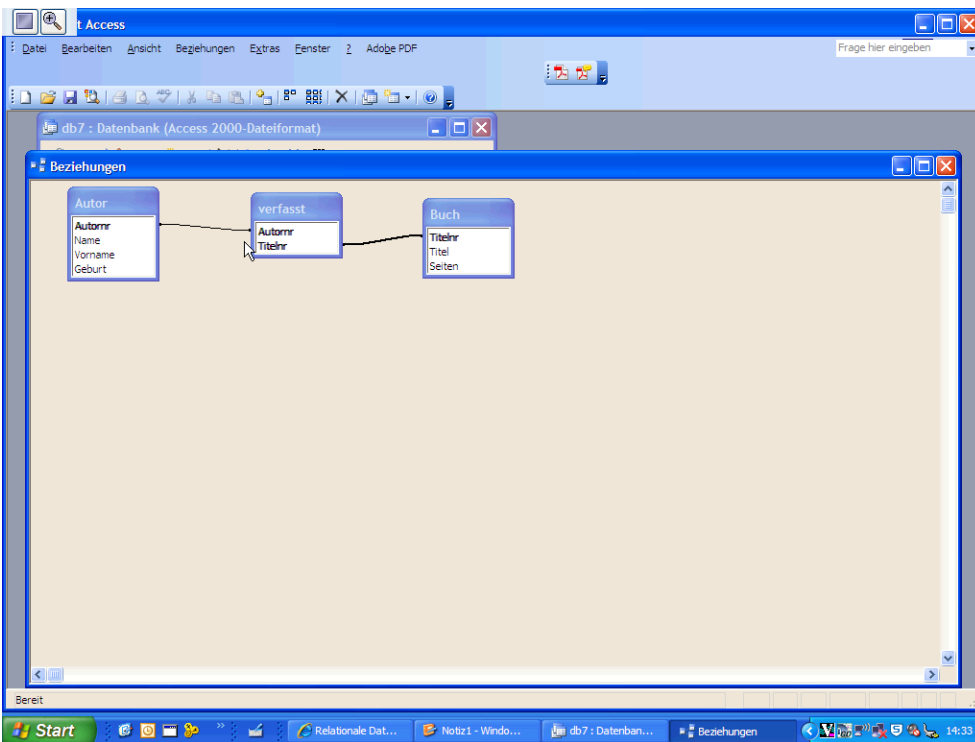
db7 : Datenbank (Access 2000-Dateiformat)

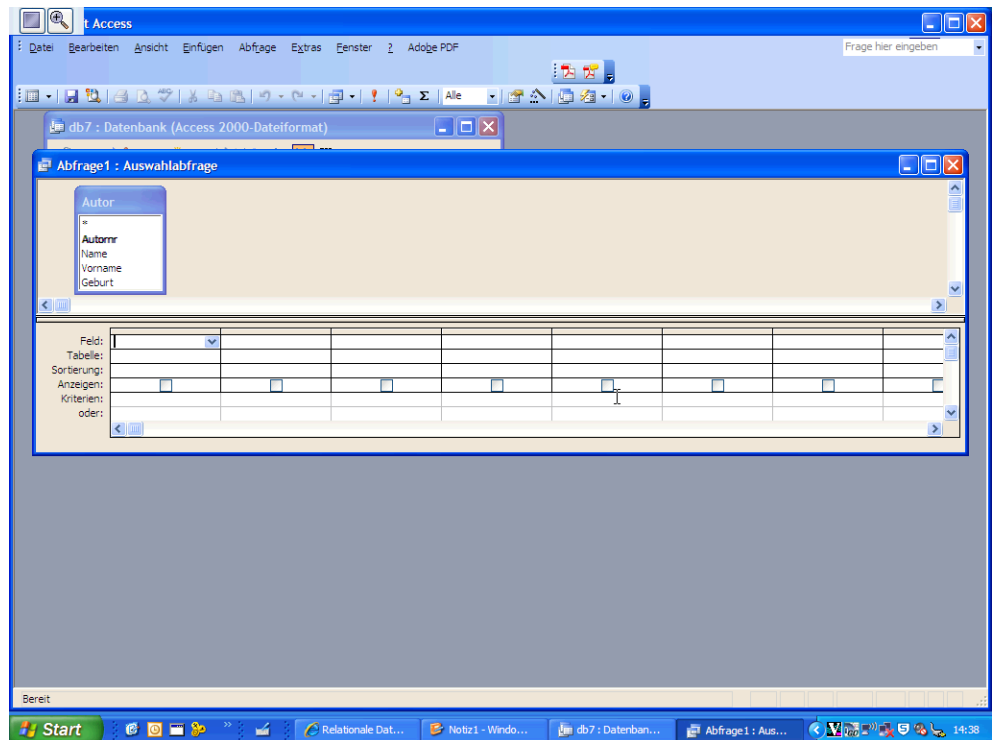
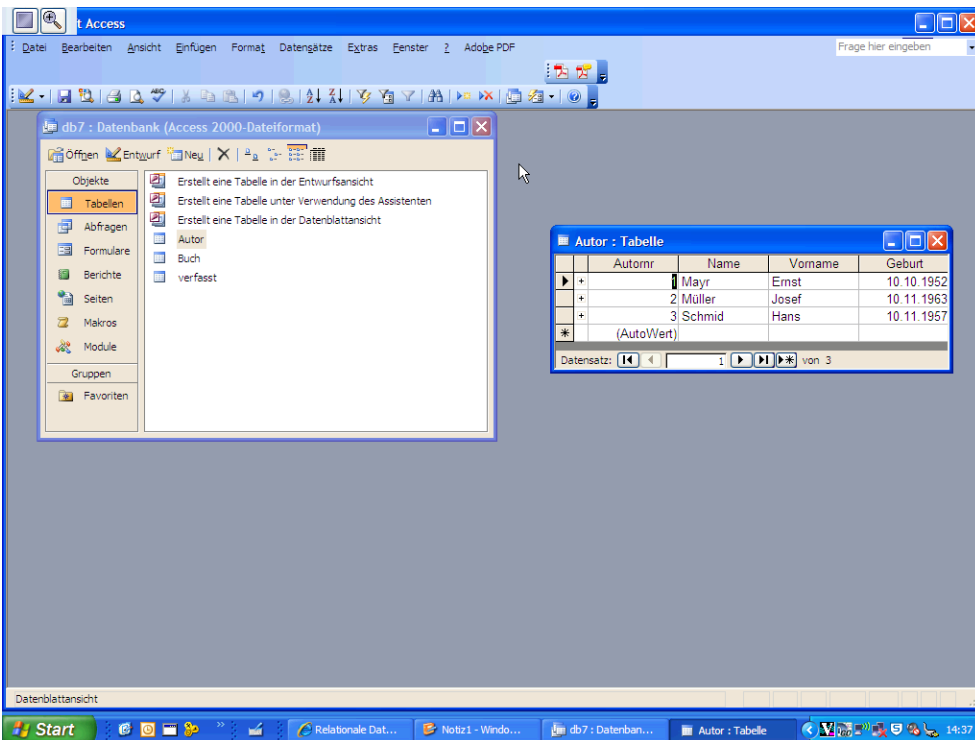
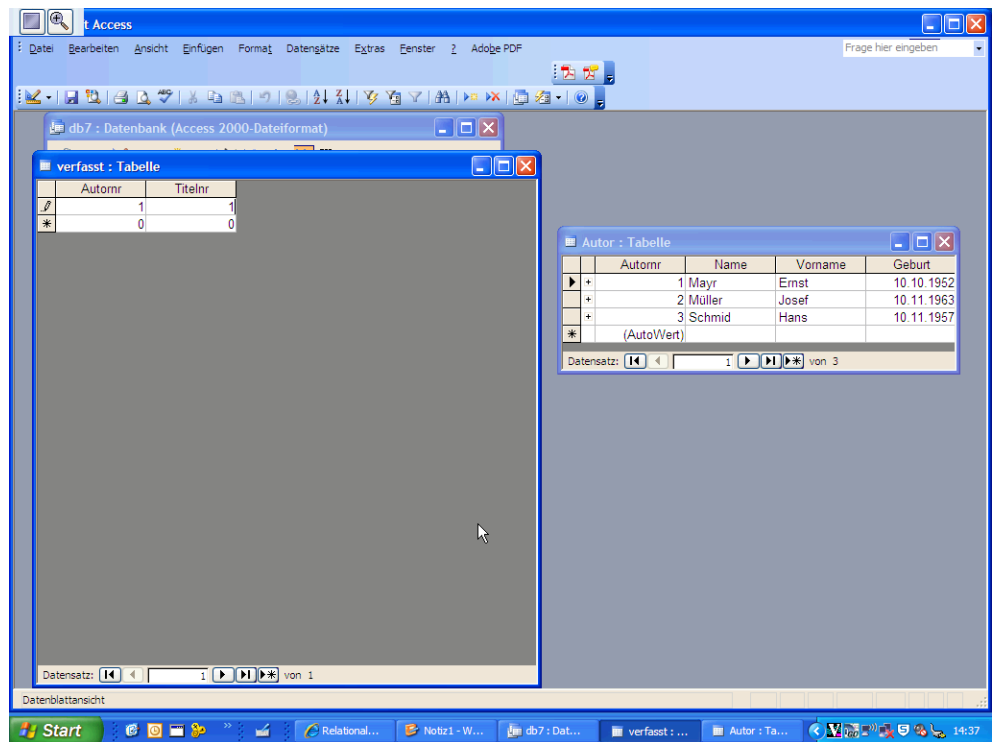
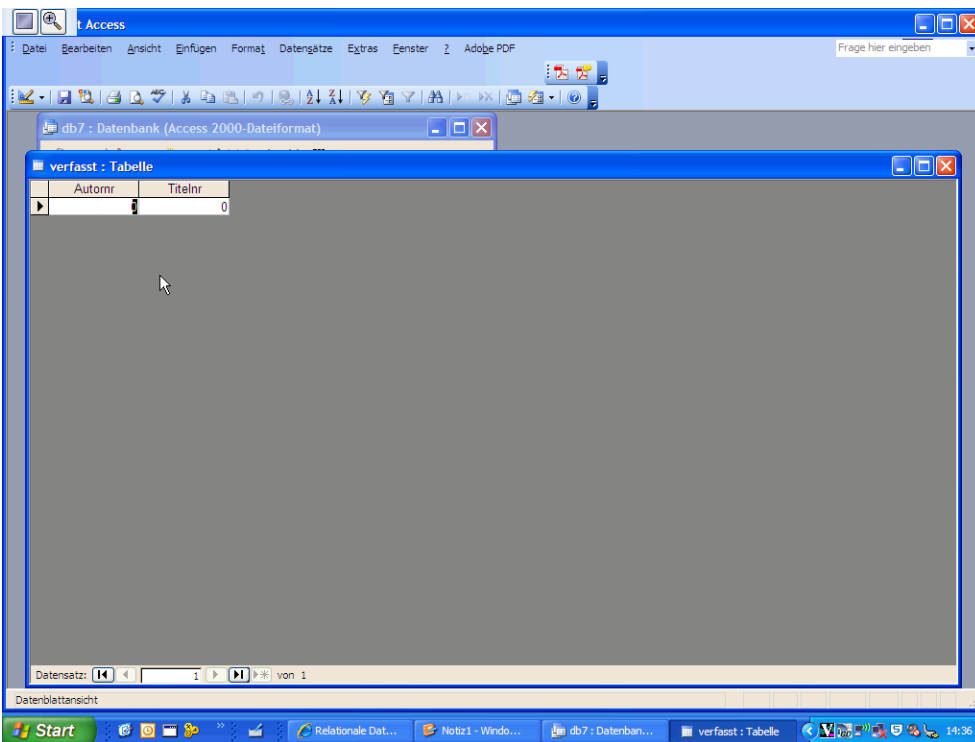
- Objekte
 - Tabellen
 - Erstellt eine Tabelle in der Entwurfsansicht
 - Erstellt eine Tabelle unter Verwendung des Assistenten
 - Erstellt eine Tabelle in der Datenblattansicht
 - Abfragen
 - Formulare
 - Berichte
 - Seiten
 - Makros
 - Module
- Gruppen
- Favoriten

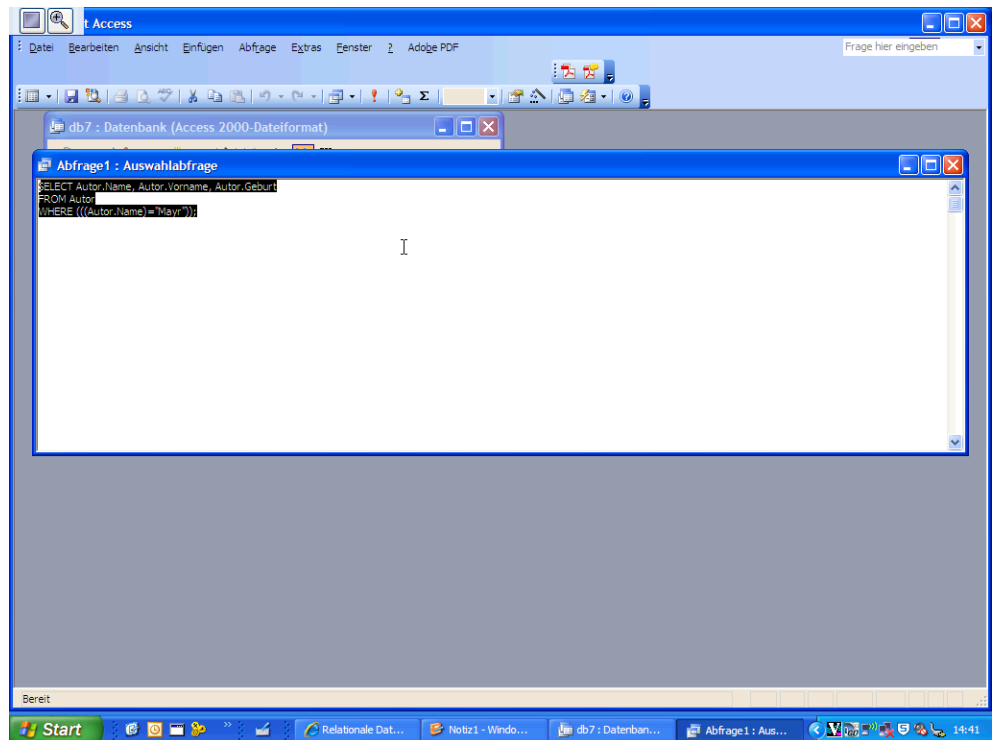
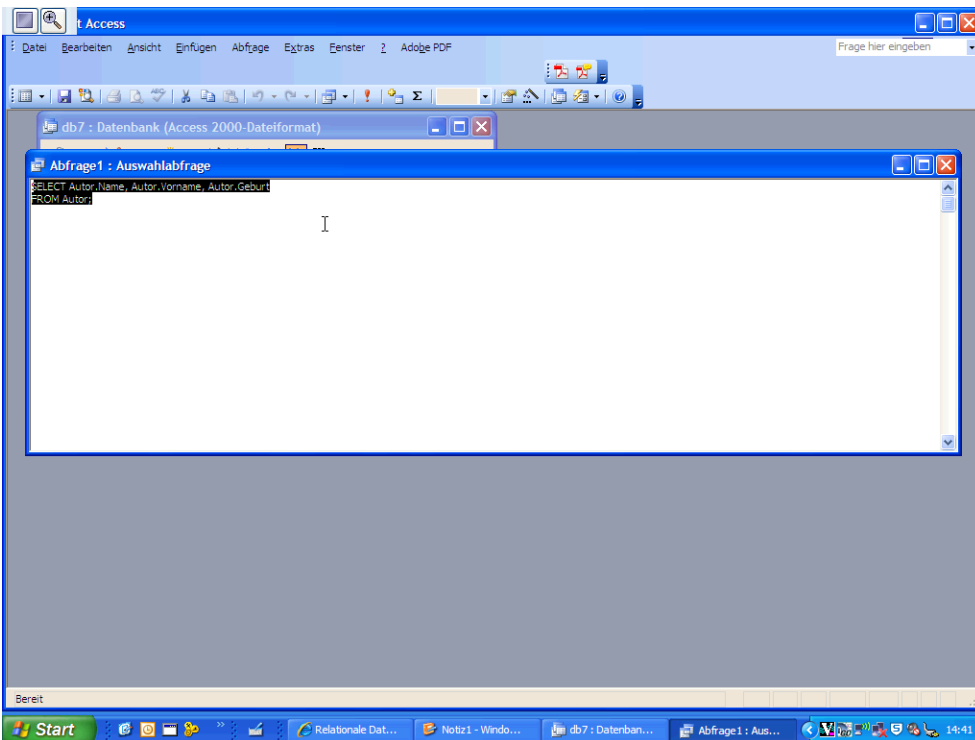
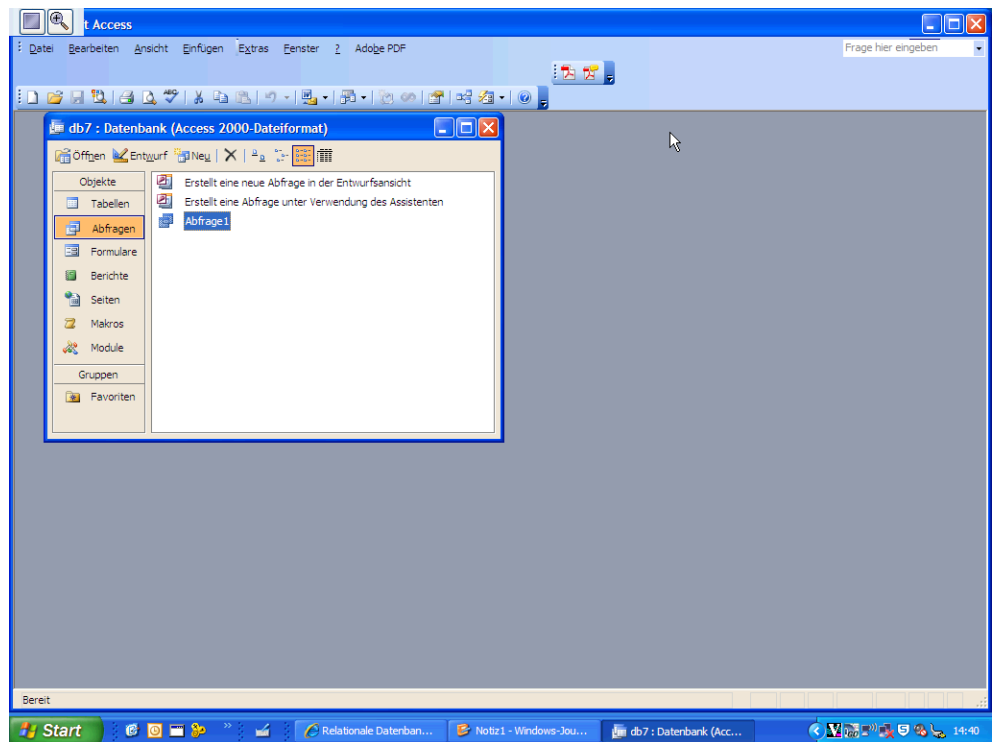
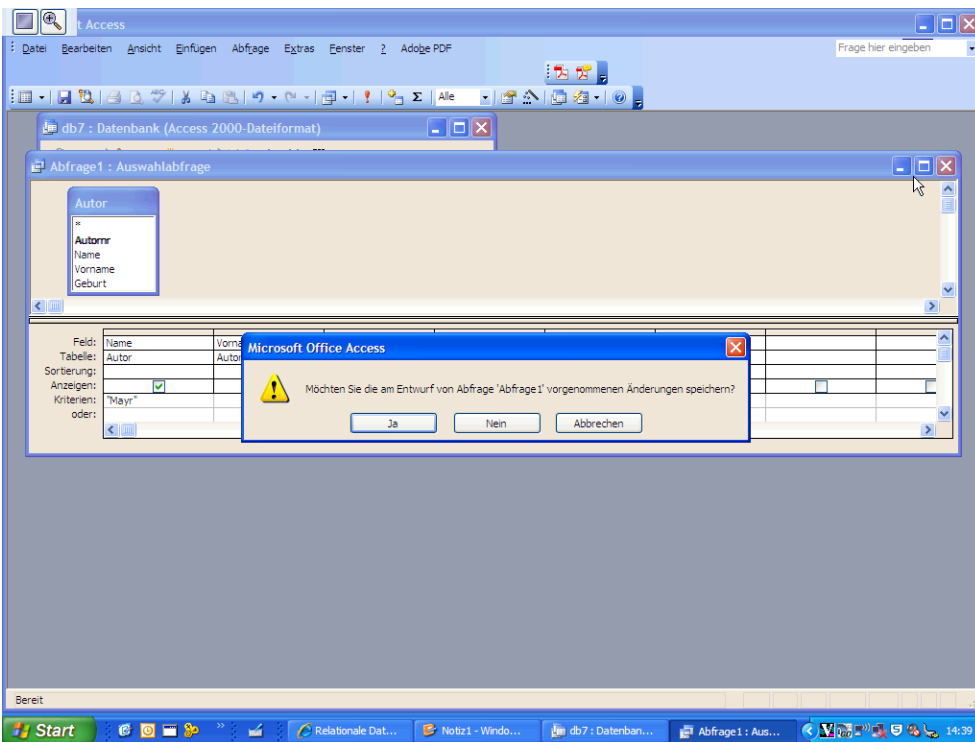
Bereit

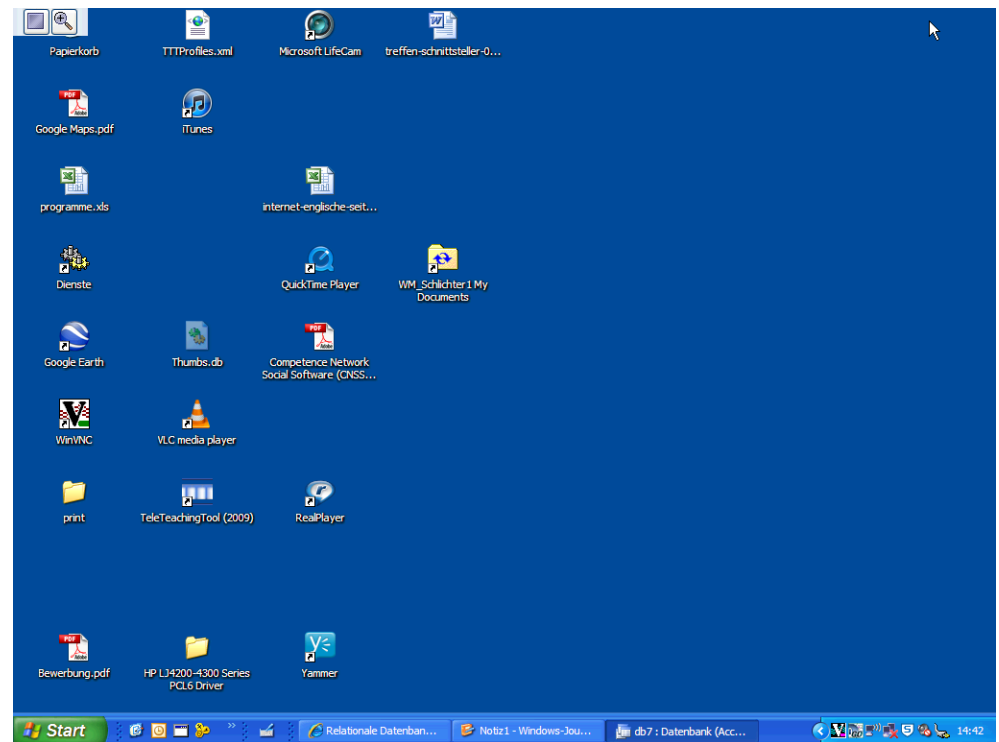
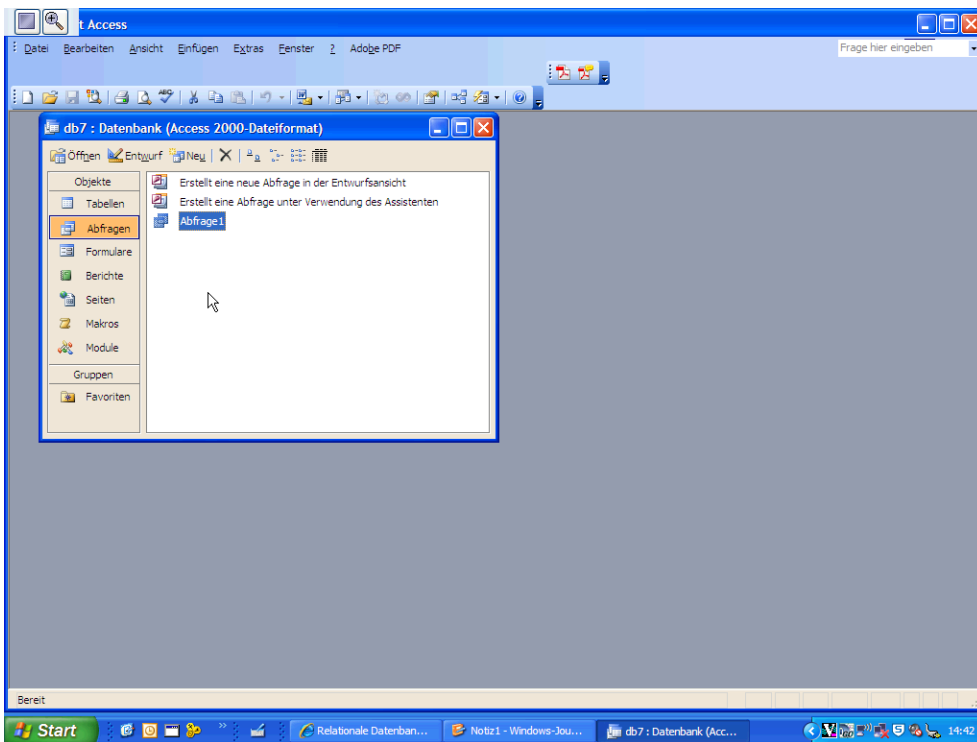
Start | Relationale Datenban... | Notiz1 - Windows-Jou... | Microsoft Access | 14:25











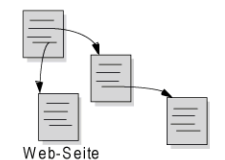
Datenbanken und Informationssysteme

Hypertext

- Fragestellungen des Abschnitts:
 - Was unterscheidet Dateisysteme von Datenbanksystemen?
 - Wie kann die Struktur der Daten in einem Datenbanksystem dargestellt werden?
 - Was sind relationale Datenbanksysteme?
 - Was sind die grundlegenden Konstrukte von HTML?

- [Dateisysteme](#)
- [Datenbanksysteme](#)
- [Datenbankentwurf](#)
- [Relationale Datenbanksysteme](#)
- [WWW - Informationssystem](#)

Hypertext = nichtlinearer Text als Netzwerk von Informationseinheiten



Querverweise zwischen Informationseinheiten (Web-Seiten, Dokumente) werden als **links** bezeichnet.
 Arten von Informationsinhalten: Text, Graphik, Bilder, Audio, Video, Animationen.
 Beschreibung der Inhalte mittels HTML (HyperText Markup Language).

Markup: Zusatzinformation wird in den Text selbst eingebettet.

```
<tagname>...text...</tagname>
Beispiel: <html>dokumentinhalt</html>
```

HTML ist eine Layoutsprache.



Das World Wide Web (WWW bzw. Web) ist ein über das Internet abrufbares Informationssystem, basierend auf dem Hypertext-Ansatz.

[Hypertext](#)

[Einführung in HTML](#)

[Cascading Style-Sheets \(CSS\)](#)

Generated by Targeteam



```

<html>
<head>
  <title>Ziel der Vorlesung</title>
</head>
<body>
<p>Diese Vorlesung gibt eine Einführung in die Informatik. Sie behandelt Themen wie</p>
<ul>
  <li>Algorithmen und Datenstrukturen</li>
  <li>Programmierung von Softwaresystemen</li>
  <li>Datenbanken</li>
  <li>Rechnernetze</li>
</ul>
</body>
</html>

```

Generated by Targeteam



HTML dient in erster Linie dazu, textuelle Information zu repräsentieren. Der Text lässt sich strukturieren, i.w. in Paragraphen, Listen und Tabellen.

ein ausführliches Tutorial stellt [selfhtml](#) bereit.

[Beispiel](#)

[Struktur-Tags](#)

[Listen](#)

[Tabellen](#)

[Grafiken](#)

[Hyperlinks](#)

Generated by Targeteam



HTML dient in erster Linie dazu, textuelle Information zu repräsentieren. Der Text lässt sich strukturieren, i.w. in Paragraphen, Listen und Tabellen.

ein ausführliches Tutorial stellt [selfhtml](#) bereit.

[Beispiel](#)

[Struktur-Tags](#)

[Listen](#)

[Tabellen](#)

[Grafiken](#)

[Hyperlinks](#)

Generated by Targeteam



Tabellen erlauben die Anordnung von Paragraphengruppen in einem zweidimensionalen Gitter.

<table> </table>: Klammerung der gesamten Tabelle.

<tr> </tr>: Klammerung einer Tabellenzeile.

<td> </td>: Klammerung einer Tabellenzelle.

<th> </th>: Klammerung einer Kopfzelle der Tabelle.

Beispiel

Attribute in <table> bzw. <td> zur Gestaltung von Tabellen, z.B.

<table border="..">: Dicke des sichtbaren Rahmens.

<table cellpadding="..">: Abstand des Zellinhalts vom Rand der Tabellenzelle.

<table width="..">: bestimmt die Breite der gesamten Tabelle.

<td align=".." valign="..">: horizontale und vertikale Ausrichtung des Inhalts innerhalb einer Zelle.

Generated by Targeteam



```
<table>
<tr>
  <th>Kopfzeile / 1. Spalte</th>
  <th>Kopfzeile / 2. Spalte</th>
  <th>Kopfzeile / 3. Spalte</th>
</tr>
<tr>
  <td>1. Zeile / 1. Spalte</td>
  <td>1. Zeile / 2. Spalte</td>
  <td>1. Zeile / 3. Spalte</td>
</tr>
<tr>
  <td>2. Zeile / 1. Spalte</td>
  <td>2. Zeile / 2. Spalte</td>
  <td>2. Zeile / 3. Spalte</td>
</tr>
</table>
```

<th>..	<th>..	<th>..
<td>..	<td>..	<td>..
<td>..	<td>..	<td>..



Beispiel



```
<th>Kopfzeile / 1. Spalte</th>
<th>Kopfzeile / 2. Spalte</th>
<th>Kopfzeile / 3. Spalte</th>
</tr>
<tr>
  <td>1. Zeile / 1. Spalte</td>
  <td>1. Zeile / 2. Spalte</td>
  <td>1. Zeile / 3. Spalte</td>
</tr>
<tr>
  <td>2. Zeile / 1. Spalte</td>
  <td>2. Zeile / 2. Spalte</td>
  <td>2. Zeile / 3. Spalte</td>
</tr>
</table>
```

<th>..	<th>..	<th>..
<td>..	<td>..	<td>..
<td>..	<td>..	<td>..

Generated by Targeteam



Beispiel



```
<table>
<tr>
  <th>Kopfzeile / 1. Spalte</th>
  <th>Kopfzeile / 2. Spalte</th>
  <th>Kopfzeile / 3. Spalte</th>
</tr>
<tr>
  <td>1. Zeile / 1. Spalte</td>
  <td>1. Zeile / 2. Spalte</td>
  <td>1. Zeile / 3. Spalte</td>
</tr>
<tr>
  <td>2. Zeile / 1. Spalte</td>
  <td>2. Zeile / 2. Spalte</td>
  <td>2. Zeile / 3. Spalte</td>
</tr>
</table>
```

<th>..	<th>..	<th>..
<td>..	<td>..	<td>..
<td>..	<td>..	<td>..

Überschrift
1. Zeile
u. u.

Generated by Targeteam



Tabellen erlauben die Anordnung von Paragraphengruppen in einem zweidimensionalen Gitter.

`<table>` `</table>`: Klammerung der gesamten Tabelle.

`<tr>` `</tr>`: Klammerung einer Tabellenzeile.

`<td>` `</td>`: Klammerung einer Tabellenzelle.

`<th>` `</th>`: Klammerung einer Kopfzelle der Tabelle.

Beispiel

Attribute in `<table>` bzw. `<td>` zur Gestaltung von Tabellen, z.B.

`<table border="..">`: Dicke des sichtbaren Rahmens.

`<table cellpadding="..">`: Abstand des Zellinhalts vom Rand der Tabellenzelle.

`<table width="..">`: bestimmt die Breite der gesamten Tabelle.

`<td align=".." valign="..">`: horizontale und vertikale Ausrichtung des Inhalts innerhalb einer Zelle.

Generated by Targeteam



HTML dient in erster Linie dazu, textuelle Information zu repräsentieren. Der Text lässt sich strukturieren, i.w. in Paragraphen, Listen und Tabellen.

ein ausführliches Tutorial stellt [selfhtml](#) bereit.

[Beispiel](#)

[Struktur-Tags](#)

[Listen](#)

[Tabellen](#)

[Grafiken](#)

[Hyperlinks](#)

Generated by Targeteam



Sie sind das wesentliche Element, um Querverweise zwischen HTML-Dokumenten zu definieren. Hyperlinks werden im Text eingebettet.

Markieren des Verweiszziels

`Text` und hier weiterer Text

Verweise innerhalb des gleichen Dokuments

`Text des Hyperlinks` und hier weiterer Text

Verweise auf andere lokale Dokumente

`Text des Hyperlinks` und hier weiterer Text

Verweise auf andere nicht lokale Dokumente

`Text des Hyperlinks` und hier weiterer Text

Generated by Targeteam



Festlegung der Formateigenschaften von HTML-Tags

interpretieren durch den Browser zum Formatieren und Positionieren von HTML-Elementen auf dem Nutzerrechner.

Einbettung von CSS-Formaten in die Web-Seite oder als separate Datei, die von Web-Seite referenziert wird.

Eigenschaften werden mit Hilfe von Regeln spezifiziert, eine Regel besteht aus Selektor, der Eigenschaft sowie dem ihr zugewiesenen Wert.

Beispiel

```
BODY { font-size: 20px;
        font-family: Helvetica;
        margin-left: 0.5em }
TD { font-family: inherit; font-size: 20px;
      vertical-align: top; }
P { font-family: Helvetica;
     font-size: 20px;
     margin-bottom: 0.5em; margin-top: 0em; }
H1 { font-family: Helvetica;
     font-size: 24px; }
H2, H3, H4, H5, H6 { font-family: Helvetica;
                     font-size: 20px;
                     margin-bottom: 0.5em; margin-top: 0.5em; }
```



Das World Wide Web (WWW bzw. Web) ist ein über das Internet abrufbares Informationssystem, basierend auf dem Hypertext-Ansatz.

[Hypertext](#)

[Einführung in HTML](#)

[Cascading Style-Sheets \(CSS\)](#)

Generated by Targeteam



• Fragestellungen des Abschnitts:

- Aus welchen (Hardware-)Elementen setzt sich ein Rechner zusammen?
- Wie kommunizieren die einzelnen Komponenten eines Rechners?
- Wie sieht die Schnittstelle zwischen Hardware und Software aus (d.h. Maschinenbefehle)?
- Wie werden Zahlen, Text, Bilder, und Töne intern dargestellt?

[Aufbau eines Rechners](#)

[Maschinenbefehle](#)

[Befehlszyklus](#)

[Interdarstellung von Information](#)

Generated by Targeteam



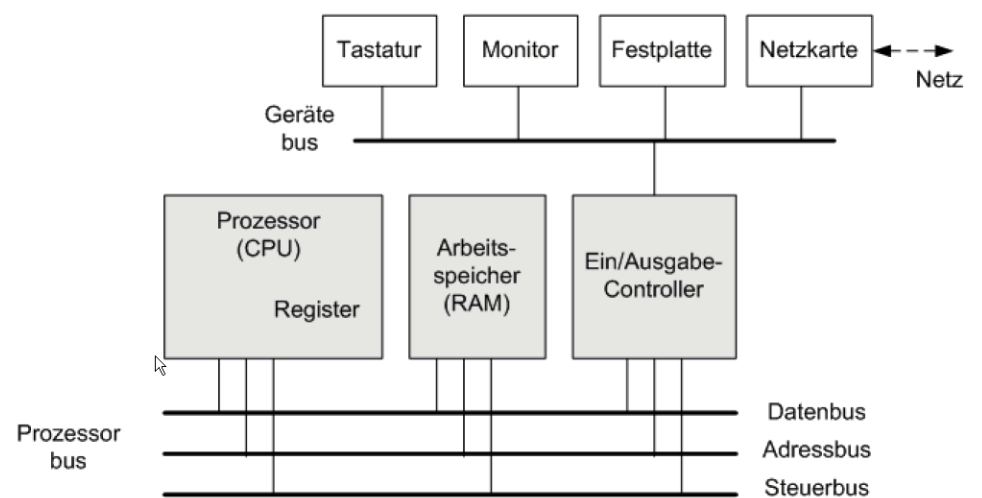
Hardware eines Computersystems: Rechner und Peripheriegeräte, z.B. Tastatur, Maus, Bildschirm, Drucker, Scanner. Sehr unterschiedliche Leistungsfähigkeit.

Beispiel eines Supercomputers



[Struktureller Aufbau eines Rechners](#)

Generated by Targeteam



[Komponenten eines Rechners](#)

[Busse zur Kommunikation](#)

[Kommunikation zwischen Arbeitsspeicher und CPU](#)

Generated by Targeteam



Im Prozessor: alle Rechen- und Umformvorgänge und die Ablaufsteuerung der Programme.

moderne Prozessoren bestehen aus mehreren Kernen

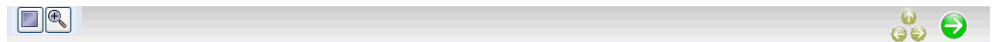
Register sind spezielle Speicherzellen im Prozessor.

Arbeitsspeicher (ASP; "main memory"): direkt adressierbare Speicherzellen der Größe 8 Bit zur Ablage von Programmen und binären Daten.

Ein/Ausgabe-Controller: setzen Nachrichten auf dem Gerätebus für Ein/Ausgabe-Geräte um, stellen Eingaben der Geräte am Bus bereit. Anschluss von Graphikkarten, Netzwerkkarten etc über den Gerätebus.

Den Ablauf der Informationsverarbeitungsvorgänge im Rechner steuern Programme.

Generated by Targeteam



Rechnerkomponenten kommunizieren über Prozessorbus und Gerätebus

Prozessorbus: schnelle Übertragung

Gerätebus (z.B. PCI): etwas langsamer, da Geräte langsamer als CPU oder Arbeitsspeicher.

Der Prozessorbus unterteilt sich in

Adressbus: zur Übergabe einer Adresse (32 oder 64 Bit)

Datenbus: zur Übermittlung von Daten (32 oder 64 Bit)

Steuerbus: mehrere Steuerleitungen

Generated by Targeteam

Kommunikation zwischen Arbeitsspeicher und CPU

CPU und Arbeitsspeicher arbeiten nicht synchron ⇒ Koordination notwendig.

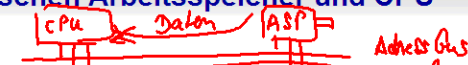
Beispiel: Übertragung von Daten aus dem Arbeitsspeicher zum Prozessor (CPU).

1. CPU legt die ASP-Adresse des gewünschten Datenelements auf den Adressbus.
2. CPU legt das Signal "Lesen" auf eine bestimmte Leitung im Steuerbus.
3. CPU legt das Signal "Adresse gültig" auf eine bestimmte Leitung im Steuerbus.
4. ASP ist passiv, "hört" aber auf Adress- und Steuerbus mit.
5. ASP erkennt, dass eine Adresse gültig ist und prüft, ob er mit dieser Adresse gemeint ist. Falls ja, führe Schritte 6-12 aus.
6. ASP prüft das Transportrichtungssignal im Steuerbus (es erkennt das Signal "Lesen").
7. ASP sucht die adressierte Speicherzelle und holt die Daten.
8. ASP legt die (adressierten) Daten auf den Datenbus.
9. ASP legt das Signal "Daten bereit" auf eine bestimmte Leitung im Steuerbus.
10. CPU hat gewartet, bis das Signal "Daten bereit" erscheint und übernimmt Daten vom Bus in das Zielregister.
11. CPU legt das Signal "Daten übernommen" auf eine bestimmte Leitung im Steuerbus.
12. ASP deaktiviert daraufhin den Datenbus und das Signal "Daten bereit".
13. Sobald das Signal "Daten bereit" vom ASP weggenommen wird, deaktiviert die CPU die von ihm verwendeten Adress- und Steuerleitungen.

Solcher Ablauf aus Aktions- und Kommunikationsschritten zwischen Kommunikationspartnern heißt "Protokoll".

Der oben geschilderte Ablauf heißt **Busprotokoll**.

Kommunikation zwischen Arbeitsspeicher und CPU



Beispiel: Übertragung von Daten aus dem Arbeitsspeicher zum Prozessor (CPU).

1. CPU legt die ASP-Adresse des gewünschten Datenelements auf den Adressbus.
2. CPU legt das Signal "Lesen" auf eine bestimmte Leitung im Steuerbus.
3. CPU legt das Signal "Adresse gültig" auf eine bestimmte Leitung im Steuerbus.
4. ASP ist passiv, "hört" aber auf Adress- und Steuerbus mit.
5. ASP erkennt, dass eine Adresse gültig ist und prüft, ob er mit dieser Adresse gemeint ist. Falls ja, führe Schritte 6-12 aus.
6. ASP prüft das Transportrichtungssignal im Steuerbus (es erkennt das Signal "Lesen").
7. ASP sucht die adressierte Speicherzelle und holt die Daten.
8. ASP legt die (adressierten) Daten auf den Datenbus.
9. ASP legt das Signal "Daten bereit" auf eine bestimmte Leitung im Steuerbus.
10. CPU hat gewartet, bis das Signal "Daten bereit" erscheint und übernimmt Daten vom Bus in das Zielregister.
11. CPU legt das Signal "Daten übernommen" auf eine bestimmte Leitung im Steuerbus.
12. ASP deaktiviert daraufhin den Datenbus und das Signal "Daten bereit".
13. Sobald das Signal "Daten bereit" vom ASP weggenommen wird, deaktiviert die CPU die von ihm verwendeten Adress- und Steuerleitungen.

Solcher Ablauf aus Aktions- und Kommunikationsschritten zwischen Kommunikationspartnern heißt

"Protokoll".

Der oben geschilderte Ablauf heißt **Busprotokoll**.

Generated by Targeteam