

# Grundlagen der EDV 5

Vorlesung mit Übungen

Dipl. Ing. Martin Ney



# Grundlagen der EDV 5

- HTML und CSS
- HTML und PHP
- CMS
- Datenbanken
- Stamm- und Bewegungsdaten
- SQL



# Internetprotokolle



- **HTTP** zum Abruf von Internetdateien von einem Server
- **HTTPS** bietet zusätzlich Verschlüsselung, wird neuerdings **zunehmend verwendet**, **gepusht** oder **vorgeschrieben**
- **FTP** zum Hochladen von Dateien auf einen Server
- **SFTP** bietet zusätzlich Verschlüsselung, wird **zunehmend verwendet**

# HTML und CSS



Die Auszeichnungssprache HTML ermöglicht es den Inhalt von der Formatierung zu trennen (analog zu Textverarbeitungsprogrammen).

In HTML wird dazu CSS (Cascading style sheets) verwendet.

Mit CSS können bestehende Formatierungs-Tags (teilweise) überschrieben oder neue Tags angelegt werden.

Es ist möglich die CSS Informationen in der HTML Datei abzulegen oder auch in einer getrennten Datei. Dann muss diese Datei per Link mit eingebunden werden.

# Übung: CSS in HTML



- Herunterladen der Datei html\_6.txt und in eine HTML-Datei kopieren (Strg-A, Strg-C, Strg-V)
- Die Datei beinhaltet im Header CSS Informationen die im Body verwendet werden
- Fügen Sie eine weitere Zeile mit der Formatierung h3 hinzu
- Ändern Sie die Farbe der Überschrift h3 und formatieren Sie alles in h3 auf linksbündig
- Ändern Sie die Farbe in h2 auf Gelb



# Übung: Formatvorlagen in Word

- Herunterladen der Textdatei Intel.txt von:  
<http://www.solarflyer.de/download/Intel.txt>
- Datei in Word öffnen oder Inhalt in eine Worddatei kopieren (Strg-A, Strg-C, Strg-V)
- Überschriften und Absätze jeweils die passende Formatvorlage zuordnen (klicken in Absatz)
- Inhaltsverzeichnis einfügen.  
Word generiert das Inhaltsverzeichnis automatisch anhand der jeweiligen Überschriften oder Unter-Überschriften





# HTML-Formulare



In HTML-Seiten können auch Daten durch den Benutzer eingegeben werden.

Hierfür werden Formulare verwendet.

```
<form method="post"  
action="formular_auswerten.html">
```

...

```
</form>
```

Die Eingaben werden an den Server gesendet der die Daten an die in **action** angegebene HTML-Datei übergibt. Darin können die eingegebenen Daten über die jeweiligen Bezeichner z.B. in php ausgewertet werden.



# Übung: HTML Formular



- Herunterladen der Datei `html_formular.txt` und in eine HTML-Datei kopieren (Strg-A, Strg-C, Strg-V)
- Die Datei beinhaltet ein HTML-Formular mit 2 Eingabefeldern.
- Erweitern Sie das Formular um ein weiteres Eingabefeld für die "body"-Hintergrundfarbe und um ein Eingabefeld für die "div"-Hintergrundfarbe.
- Benennen Sie die Schaltfläche zum Absenden um, z.B. in: "Eingabedaten absenden"
- Entfernen Sie unnötige Codeteile



# PHP



**PHP** ist eine Skriptsprache mit einer an Perl und C angelehnten Syntax, die hauptsächlich zur Erstellung dynamischer Webseiten oder Webanwendungen verwendet wird.

PHP ist eine freie Software mit einer breiten Datenbankbindung.

Die **Entwicklung begann 1995**. Es ist die meistverwendete serverseitige Programmiersprache im Internet (über 82 % der Internetseiten werden damit betrieben, Tendenz steigend).

# PHP-Code



**PHP**-Code wird in eine HTML-Seite integriert. Dabei wird der Code ausgeführt bevor die HTML-Seite von dem Server ausgeliefert wird. In der dann ausgelieferten HTML-Seite befindet sich nur noch das Ergebnis des php-Codes.

PHP Codefragmente werden mit **<?php** oder nur **<?** eingeleitet und mit **?>** beendet.

```
<?php
```

```
    echo 'Hallo Welt!';
```

```
?>
```



# PHP-Code in HTML



Der **PHP**-Code wird in eine HTML-Seite an beliebiger Stelle innerhalb des HTML-Code platziert.

Beispiel:

```
<!DOCTYPE HTML>
<html>
  <head>
    <title>Hallo-Welt-Beispiel</title>
  </head>
  <body>
    <?php
      echo 'Hallo Welt!';
    ?>
  </body>
</html>
```

# PHP-Anweisungen



```
<? echo 'Hallo Welt!'; ?>
```

```
<? $back_color = #CCAAFF; ?>
```

```
<body bgcolor="<?php echo $back_color ?>">
```

```
Datum: <? echo date('d.m.Y'); ?>
```

# Übung: Formular auswerten



- Herunterladen der Datei `html_formular_auswerten.txt` und in eine **PHP-** Datei kopieren (Strg-A, Strg-C, Strg-V)
- Die Datei beinhaltet php-Code mit dessen Hilfe die Formulareingaben im HTML-Code verwendet werden können.
- Geben Sie die zusätzlich eingegebenen Farben als Text aus. (z.B. `body-Hintergrundfarbe=...`)
- Verwenden Sie die zusätzlich eingegebenen Farben jeweils als Hintergrundfarben



# Übung: Verlinken auf dem Server



- Öffnen Sie in Filezilla den FTP-Server Solarflyer
- Erstellen Sie dort ein neues Verzeichnis mit ihrem Vor- und Nachnamen, z.B.:  
martin\_ney
- Laden Sie beide Dateien auf den Server in Ihr Verzeichnis und testen Sie die Funktionalität
- Nutzen Sie die Datei  
html\_formular\_auswerten.**php** um einen Link einzubauen. Der Link soll zurück auf die Datei  
html\_formular.**php** verlinken



# Übung: Indexdatei erstellen



- Erstellen Sie eine HTML-Datei mit dem Namen `index.php` als Startdatei für Ihr Verzeichnis.
- Fügen Sie in diese Datei Text und einen Link ein. Der Link soll auf die Datei `html_formular.php` verweisen.
- Fügen Sie in der Datei `html_formular_auswerten.php` ebenfalls einen Link auf die Datei `html_formular.php` ein.
- Laden Sie die Dateien auf den Server in Ihr Verzeichnis und testen Sie die Funktionalität





# Content Management Systeme



- Content Management Systeme (CMS) verwalten den Inhalt auf einer Internetseite.
- Es werden häufig Vorlagen für den Seitenlayout verwendet, der Inhalt der Seite kommt dann aus einer Datenbank
- Der Anwender, z.B. ein Redakteur, braucht dann (normalerweise) keine HTML Kenntnisse
- Beim Aufbau des CMS und bei Anpassungen sind jedoch HTML, CSS und Datenbankkenntnisse hilfreich

# Content Management Systeme



Folgende CMS sind sehr bekannt und weit verbreitet:

- WordPress  
(basiert auf php und der Datenbank MySQL, über 50 % der Websites die mit einem CMS betrieben werden basieren auf WordPress)
- Typo3
- Joomla
- Drupal



# Datenbanken



- **Datenbanken** dienen der strukturierten und schematisierten Speicherung großer Datenmengen
- Die Datenbank besteht aus dem Datenbankmanagementsystem (DBMS) und den eigentlichen Daten
- Die Form der Speicherung wird durch das Datenbankmodell beschrieben
- Relationale Datenbanken sind das mit Abstand häufigste Datenbankmodell



# Relationale Datenbanken



- **Relationale Datenbanken** bestehen aus Tabellen und aus Beziehungen (Relationen) zwischen diesen
- Das zugehörige Datenbankmanagementsystem wird als **Relationales Datenbankmanagementsystem** bezeichnet (RDBMS)
- Zum Abfragen und Manipulieren der Daten wird überwiegend die Datenbanksprache **SQL** (Structured Query Language) eingesetzt



# Datenbanktabellen



- Datenbanktabellen bestehen aus Zeilen gleichförmiger Datensätze
- Ein Datensatz wird durch Felder (Spalten) definiert (wie Exceltabellen, jedoch typisiert)
- Die Anzahl der Felder einer Tabelle ist fest, die Anzahl der Zeilen kann beliebig sein

ID	RegionID	Region2ID	ExternID	ExternVersi	_Kategoriell	NAME
238						Afghanistan
239	6					Ägypten
240						Aland
241	2					Albanien
242						Algerien
245	2					Andorra
246	6					Angola
247						Anguilla
248						Antigua und Barbuda
249	5					Aquatorialguinea
250	5					Argentinien
251						Armenien

# Indizes in Tabellen



- Datensätze in Tabellen können mit einem oder mehreren **Indizes** versehen werden
- Ein Index kann sich auf ein Tabellenfeld oder auf mehrere Felder beziehen
- Sehr häufig wird eine natürliche (ganze) Zahl als Index verwendet. Der primäre Index erhält oft den Namen "ID"
- Weitere Indizes sind möglich, diese werden oft mit "ID\_" und einem weiteren Namen benannt, je nachdem worauf sie sich beziehen



# Indizes in Tabellen



- **Indizes** Erleichtern das Suchen und Sortieren von Datensätzen, insbesondere in großen Datenmengen
- Es gibt eindeutige Indizes, d.h. jeder Datensatz hat eine eigene Zahl und nicht eindeutige Indizes, d.h. mehreren Datensätzen ist die gleiche Zahl zugeordnet
- Jedes Datenbankfeld kann als Index verwendet werden. Indizierte Datenbankfelder werden häufig zur Verknüpfung von Tabellen verwendet



# Übung: Datenbanken 1



- Erstellen Sie mit MS ACCESS eine Datenbank
- Legen Sie darin die Tabelle Bücher mit den jeweiligen Feldern an (Entwurfsansicht)
- Füllen Sie die Felder mit Inhalt (Tabellenansicht)
- Führen Sie SQL-Abfragen auf der Tabelle aus (Abfragen)
- Exportieren Sie die Abfrageergebnisse als csv-D.





# Relationen



- Relationen zwischen zwei Tabellen stellen Beziehungen zwischen einzelnen Feldern der Tabellen dar
- Zumeist werden ID-Felder für Relationen verwendet.
- Die wichtigsten Arten von Relationen sind:
  - 1:1 Beziehung
  - 1:n Beziehung
  - m:n Beziehung

# Relationen Beispiel



- Beispiel für eine 1:n Beziehung:
  - Kunde – Aufträge  
(ein Kunde kann mehrere Aufträge haben, aber ein Auftrag ist genau einem Kunden zugeordnet)
  - Kunde - Kontakt
  - Länder – Städte (jede Stadt liegt in einem Land)
- Beispiel für eine m:n Beziehung
  - Kunde - Auto (bei einer Autovermietung)  
(ein Kunde kann mal das eine oder das andere Auto mieten, ebenso kann jedes Auto von verschiedenen Kunden gemietet werden)

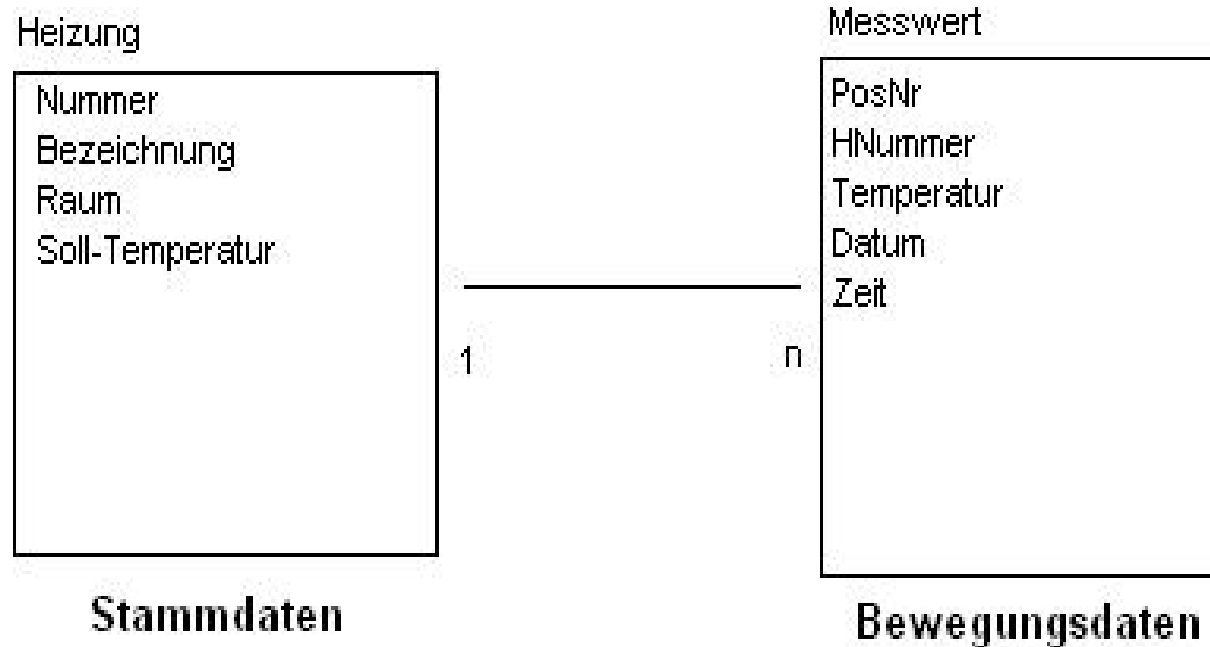


# Stamm- und Bewegungsdaten

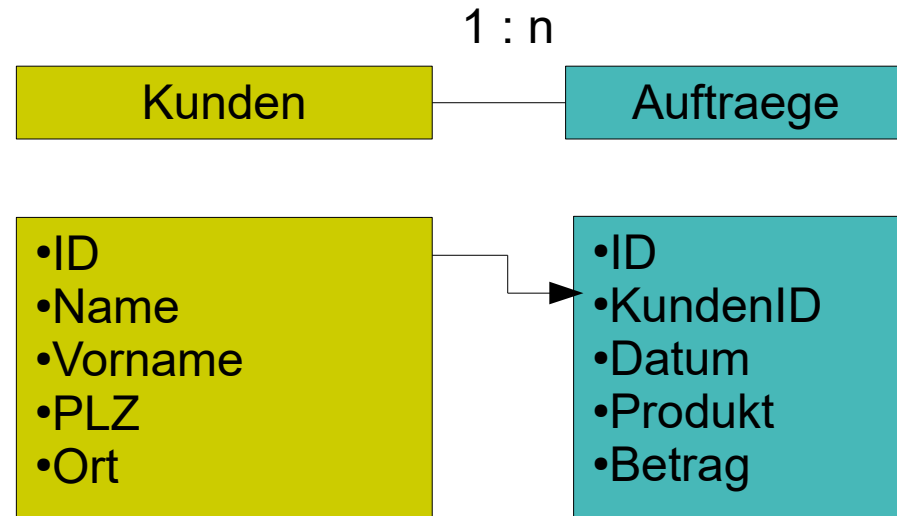


- Stammdaten beinhalten Grundinformationen z.B. über betriebliche Objekte
- Bewegungsdaten ändern sich häufiger
- Definition z.B. über existentielle Abhängigkeiten oder Änderungsvolumen
- Z.B. Kunden sind Stammdaten, Aufträge, Preise oder Rechnungen sind Bewegungsdaten
- Beispiele: **Heise-Preisdatenbank** oder **Amazon Produktbewertungen** oder Bankkonten

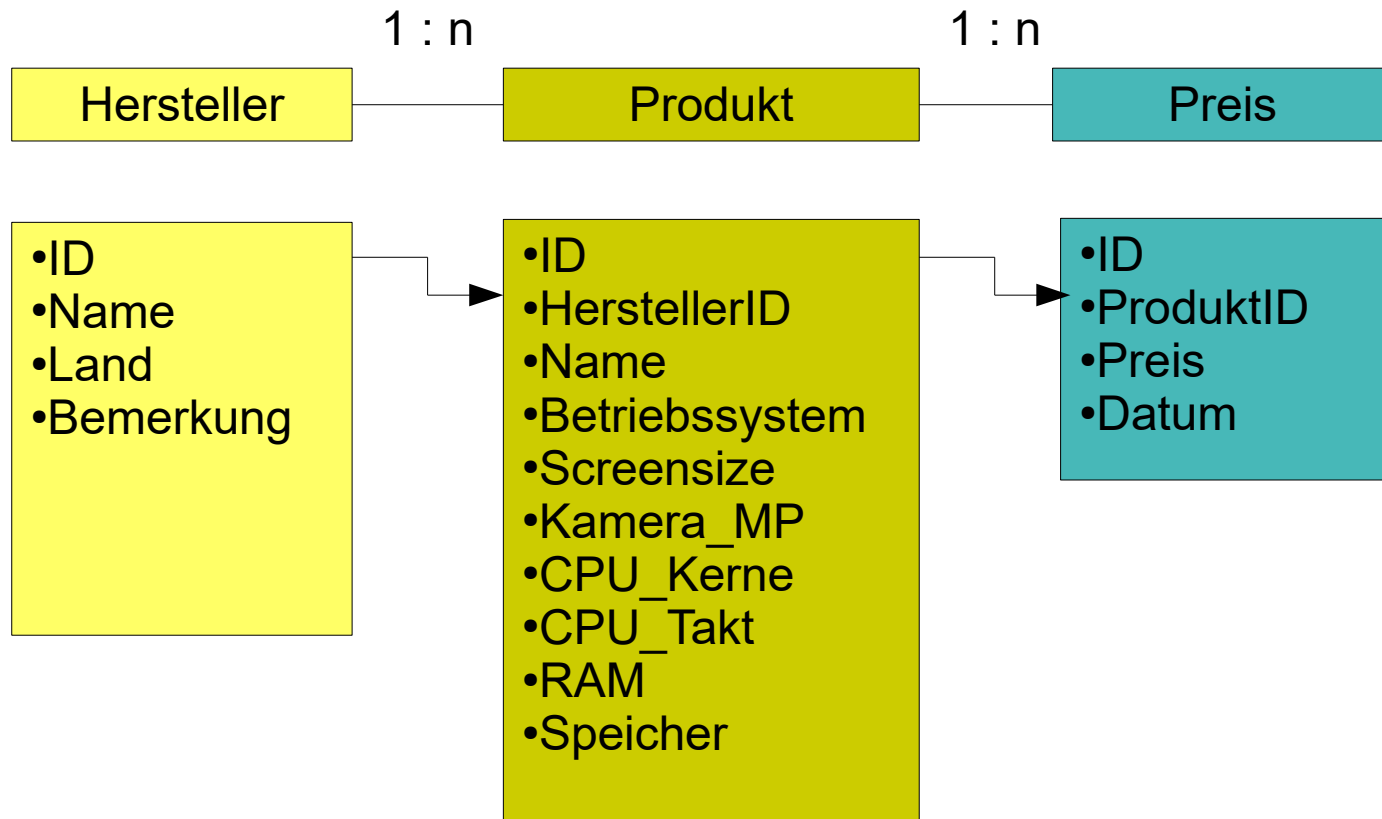
# Beispiel zu Stamm- und Bewegungsdaten: Heizungen



# Beispiel zu Stamm- und Bewegungsdaten: Aufträge



# Beispiel zu Stamm- und Bewegungsdaten: Handys



# SQL



- Datenbanksprache **SQL** (Structured Query Language) dient dem Manipulieren oder Abfragen von Daten
- Manipulieren der Struktur geht mit **CREATE TABLE**
- Einfügen von Daten geht mit **INSERT**
- Ändern von Daten geht mit **UPDATE**
- Auslesen von Daten mit **SELECT**
  - Beispiel: **SELECT \* FROM** Kundentabelle

# SQL Befehl SELECT



- Mit dem SQL Befehl **SELECT** können alle Daten einer Tabelle oder selektive Daten einer oder mehrerer Tabellen abgefragt werden
- Bei der Abfrage können die Daten auch gruppiert oder sortiert werden
- Einschränkung der auszugebenden Felder:
  - **SELECT** Name, Vorname, Ort **FROM** Tabelle
- Einschränkung der Datensätze:
  - **SELECT** Name **FROM** Tabelle **WHERE** PLZ=55100



# SQL Befehl SELECT



- Verknüpfung über 2 Tabellen hinweg:  
"Gib mir den Kundennamen, Kundenort und den Betrag aller Aufträge im Jahr 2016"
- **SELECT** Name, Ort, Betrag  
**FROM** Kunden, Auftraege  
**WHERE** Auftrag.KundenID = Kunden.ID  
**AND** YEAR(Auftraege.Datum) = 2016

(Die Funktion YEAR ermittelt aus einem Datum das Jahr)



# Übung: Datenbanken 2



- Erstellen Sie mit MS ACCESS eine Datenbank
- Legen Sie darin die beiden Tabellen Kunden und Auftraege mit den jeweiligen Feldern an (Entwurfsansicht)
- Füllen Sie die Felder mit Inhalt (Tabellenansicht)
- Führen Sie SQL-Abfragen auf der Tabelle aus (Abfragen)
- Exportieren Sie die Abfrageergebnisse als csv-D.



# Customer Relationship Management



- **Customer Relationship Management Systeme (CRM)** dienen der Kundenpflege. Darin werden alle Daten verwaltet die in der Kundenbeziehung von belang sind, z.B. alle Kontakte, mit Datum, Betreff und Kontaktpersonen.
- Bei einem Kundenkontakt werden in einer Übersicht alle bisher angefallenen Kontakte, z.B. Anrufe, E-Mails oder Kundenbesuche angezeigt.
- Die Kontakthistorie hilft auf die Kundenbedürfnisse ein zu gehen



# Übung: Datenbanken 3



- Erweitern Sie die MS ACCESS Datenbank aus der vorhergehenden Übung mit Kundentabelle und Auftragstabelle zu einem kleinen CRM: Fügen Sie eine Tabelle für Kundenkontakte hinzu
- Tabelle anlegen und Felder definieren in der Entwurfsansicht
- Füllen der Felder mit Inhalt in der Tabellenansicht
- Führen Sie SQL-Abfragen auf der Tabelle aus (Abfragen)
- Exportieren Sie die Abfrageergebnisse als csv-D.

