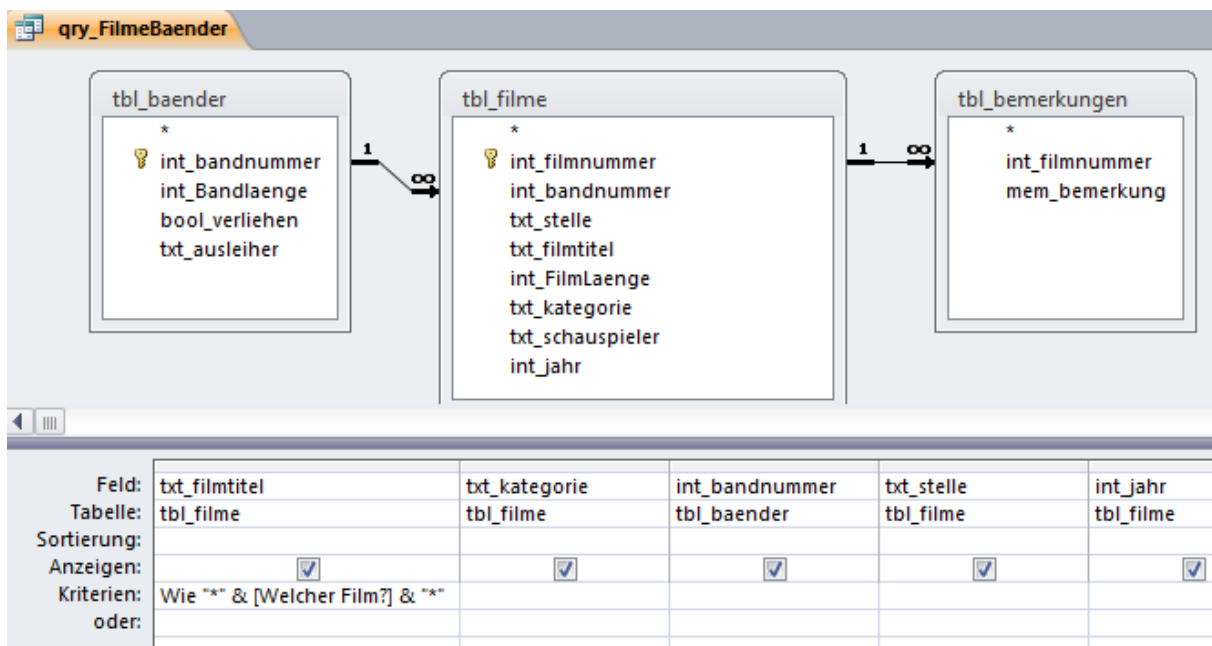


# Handbuch zum Selbststudium und für Schulungskurse

David Niegisch  
Oliver Buckenhofer  
Alexandra von Cube

# ACCESS 2010 FÜR EINSTEIGER



Dieses Dokument wird unter folgender creative commons veröffentlicht:  
<http://creativecommons.org/licenses/by-nc-sa/3.0/>

Januar 2011 Version 3.0



## Inhaltsverzeichnis

<b>1</b>	<b>Einführung.....</b>	<b>1</b>
1.1	Über die Autoren .....	1
1.2	Was ist eine Datenbank?.....	1
1.3	Manuelle Datenbanken .....	1
1.4	EDV-gestützte Datenbanken .....	1
1.5	Die Tabelle in der Datenbank .....	2
1.6	Datensatzaufbau.....	3
1.7	Relationale Datenbank.....	4
1.8	Was sind Primärschlüsselfelder und Fremdschlüsselfelder?.....	5
1.8.1	Primärschlüsselfeld .....	5
1.8.2	Fremdschlüsselfeld.....	5
1.9	Beziehungen .....	5
1.9.1	Die 1:n-Beziehung.....	6
1.9.2	Die n : m-Beziehung .....	7
1.9.3	Die 1 : 1-Beziehung.....	8
1.10	Referentielle Integrität .....	8
1.11	Die Normalformen .....	9
1.11.1	Erste Normalform (1. NF).....	9
1.11.2	Zweite Normalform (2. NF):.....	11
1.11.3	Dritte Normalform (3. NF).....	11
1.12	Indizes .....	12
<b>2</b>	<b>Die Benutzeroberfläche von Access 2010 .....</b>	<b>13</b>
2.1	MS Office 2010 .....	13
2.2	Die Oberfläche nach dem Start von Access 2010.....	13
2.3	Was es sonst noch so gibt.....	17
<b>3</b>	<b>Die erste Tabelle .....</b>	<b>19</b>
3.1	Vorüberlegung .....	19
3.1.1	Entity Relationship Model (ERM) .....	19
3.1.2	Aus wie vielen Tabellen soll die Datenbank bestehen? .....	20
3.1.3	Die Entwurfsansicht: Feldnamen, Felddatentyp und Feldgröße: .....	21
3.1.4	Feldeigenschaften.....	24
3.2	Ungarische Notation .....	31
3.3	Eingabe von Feldnamen und ihren Eigenschaften .....	32
3.4	Bestimmung und Eingabe der Feldeigenschaften .....	35
3.4.1	Eingabe von Daten .....	38
3.4.2	Ändern der Datenstruktur.....	39
3.4.3	Bewegen in den Tabellen (Navigationssymbole und Tastaturbelegung).....	40
3.4.4	Datensätze in der Tabellenansicht suchen.....	41
3.5	Tabellen verknüpfen .....	43
<b>4</b>	<b>Abfragen.....</b>	<b>45</b>
4.1	Trennen von Daten und Pflege derselben.....	45
4.2	Was leisten Abfragen?.....	47
4.3	Wie erstellt man eine Abfrage? .....	48
4.3.1	Allgemeines.....	48
4.3.2	Eine Abfrage und ihr Dynaset - Eine Frage und ihre Antwort.....	50
4.3.3	Spezielles: Beispiele, Übungen und Tricks.....	50
4.3.4	Komplexes.....	58
4.4	Serienbriefe aus Access-Daten erstellen.....	64
4.5	Andere Abfragen .....	65
4.5.1	Die Tabellenerstellungsabfrage.....	65

4.5.2	Die Aktualisierungsabfrage .....	66
4.5.3	Die Löschartfrage .....	67
4.5.4	Die Anfügeabfrage .....	67
<b>5</b>	<b>Formulare erstellen .....</b>	<b>69</b>
5.1	Formulare mit dem Auto-Formular erstellen.....	69
5.2	Arbeiten mit Formularen .....	70
5.2.1	Bewegen in Formularen .....	70
5.2.2	Datensätze hinzufügen .....	71
5.2.3	Datensätze löschen .....	71
5.2.4	Speicherung von Eingaben und Änderungen .....	71
5.2.5	Änderungen rückgängig machen.....	71
5.2.6	Suchen und Ersetzen .....	72
5.3	Ändern eines Formularlayouts.....	72
5.3.1	Formular in der Layoutansicht .....	72
5.3.2	Formular in der Entwurfsansicht.....	73
5.3.3	Anpassen der Überschrift .....	74
5.3.4	Markieren von Steuerelementen.....	75
5.3.5	Verschieben von Steuerelementen .....	77
<b>6</b>	<b>Berichte - Drucken in Access .....</b>	<b>79</b>
6.1	Listendruck mit dem Berichtsassistenten .....	79
6.1.1	Der Assistent, was leistet er .....	79
6.1.2	Text ändern und einfügen - Berichtskopf und Seitenkopf .....	84

# 1 Einführung

## 1.1 Über die Autoren

Oliver Buckenhofer ist einer der Geschäftsführer der Firma Campus Training Center GmbH und ist als Dozent im Bereich Microsoft Office tätig. Er kennt Microsoft Excel seit der ersten Version. Bei Fragen und allgemeinen Feedback zu diesem Skript erreichen Sie ihn über seine Homepage:

<http://www.campus-training-center.de/>

oder über seine E-Mail-Adresse: [buckenhofer@campus-training-center.de](mailto:buckenhofer@campus-training-center.de)

David Niegisch ist Freier Mitarbeiter bei der Firma Campus Training Center GmbH und ebenfalls als Dozent im Bereich Microsoft Office tätig, insbesondere für Access und Excel. Sie erreichen ihn über seine E-Mail-Adresse: [niegisch@yahg.net](mailto:niegisch@yahg.net)

Ganz herzlich möchten die Autoren Frau Alexandra von Cube vom Rechenzentrum der Universität Dortmund und Frau Christine Damberg danken, die eine ausgezeichnete Vorgängerversion für die Neuentwicklung dieses Skripts zur Verfügung gestellt haben.

## 1.2 Was ist eine Datenbank?

Eine Datenbank ist eine Ansammlung von Informationen, die sich auf ein bestimmtes Thema oder einen bestimmten Zweck beziehen, z. B.: Bücher in einer Bibliothek, Waren im Warenlager oder Gehaltsabrechnungen im Betrieb etc., die verwaltet werden müssen.

In der Datenverarbeitung spricht man von einem Datenbanksystem, wenn dieses aus einer Datensammlung und einem Verwaltungssystem besteht. Die Datensammlung ist nach einer bestimmten Struktur aufgebaut und das Verwaltungssystem organisiert die Zugriffe auf die Datenbank und deren Bestand. Was aber bedeutet "...welches die Zugriffe auf die Datenbank und deren Bestand organisiert..." genauer?

Es bedeutet nichts anderes, als dass dieses Verwaltungssystem dem Nutzer ermöglicht, bestimmte Informationen (Daten) zu erfassen, zu speichern, wieder aufzufinden, zu aktualisieren oder weitere Operationen mit den Daten durchzuführen, wie z. B. das Sortieren nach bestimmten Kriterien.

## 1.3 Manuelle Datenbanken

Ein Beispiel für manuelle Datenbanken sind Karteikästen. Das Auffinden der Daten wird hier durch zusätzliche Register erleichtert. Ein Karteikasten enthält zusammengehörende, nach bestimmten Kriterien geordnete Daten. Jede Karte enthält i. d. R. die gleiche, festgelegte Anzahl von Eintragungen. Der Benutzer erfasst seine Daten manuell auf einer Karteikarte und ordnet diese nach dem von ihm bestimmten Kriterien in seinen Karteikasten. Um Änderungen vornehmen zu können, muss er die entsprechende Karte aus dem Karteikasten herausuchen, die Daten ändern und die Karte dann wieder an seinen Platz zurückstellen.

## 1.4 EDV-gestützte Datenbanken

Wie schon erwähnt, besteht eine EDV-gestützte Datenbank aus einer Datensammlung und einem Verwaltungsprogramm, das diese Datensammlung nach bestimmten Kriterien organisiert und kontrolliert.

Im Vergleich mit den oben genannten Datenbanken weisen EDV-gestützte Datenbanken erhebliche Vorteile auf. Sie erlauben

- bequemere Erfassung der Daten,
- schnelleres Finden,
- einfachere, z. T. automatische Aktualisierung der Daten,
- Export der Daten in andere Formate,
- flexible, datenbankübergreifende Anordnung und Zusammenstellung von Daten, somit das
- Aufdecken bisher unbekannter Informationszusammenhänge und damit eine
- effektivere Auswertung der Daten.

Als Ergebnis kann man folgende Vergleiche und Erklärungen für herkömmliche Datenbanken und EDV-gestützte Datenbanken aufstellen:

Karteikastensystem



Datenbanksystem

Ein Karteikastensystem enthält zusammengehörende, nach bestimmten Kriterien geordnete Karteien wie ein Datenbanksystem nur mit dem Unterschied, dass ein Datenbanksystem zusätzlich noch ein Verwaltungsprogramm besitzt.

Karteikasten



Tabelle

Jede Karte enthält i. d. R. die gleiche festgelegte Anzahl von Eintragungen. Ähnlich fasst eine Tabelle eine Datenmenge unter einem gemeinsamen Oberbegriff, dem Tabellennamen, zusammen.

Karteikarte



Datensatz  
(Tabellenzeile)

Jede Karteikarte entspricht einem Datensatz. Als Beispiel steht auf einer Karte folgende Angabe:

*2, Müller, Michael 50939 Köln.*

Genauso enthält dementsprechend ein Datensatz die genannten Informationen.

Eintragungsfeld



Datenfeld/Attribut  
(Tabellenspalte)

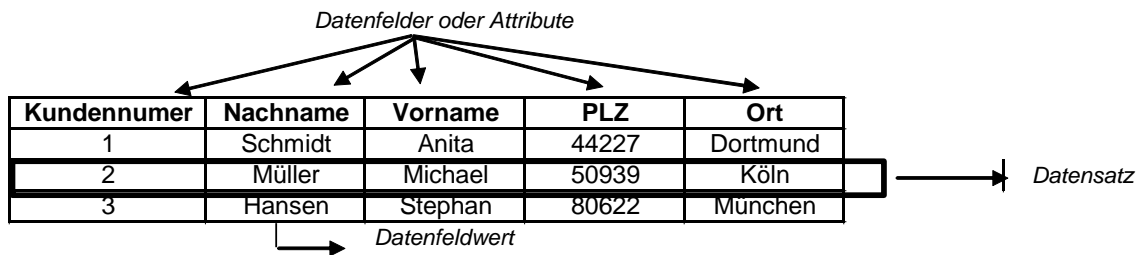
Für jede Eintragung auf der Karte steht ein bestimmtes Feld bzw. ein bestimmter Bereich auf der Karte zur Verfügung. In Tabellen sind diese Felder durch die Tabellenspalten voneinander abgegrenzt.

## 1.5 Die Tabelle in der Datenbank

Jede Datenbank besteht aus beliebig vielen, gleichartigen Datensätzen, wobei ein Datensatz einer Tabellenzeile entspricht. Jeder Datensatz besteht aus einer bestimmten Anzahl von Datenfeldern, hier entspricht jedes Feld einer Tabellenspalte. Es ergibt sich somit insgesamt ein tabellenartiger Aufbau.

Die folgende Abbildung zeigt die vereinfachte Darstellung der Datenbankdatei „Kunde“ in Form einer Tabelle.

**Tabelle Kunden**



**Abd. 1.1 Datenbankinformationen in Tabellenform**

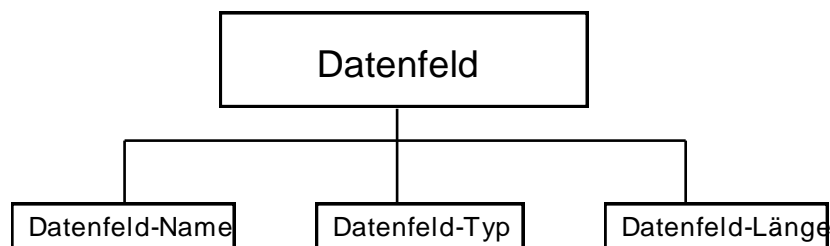
Die Tabelle 'Kunden' hat fünf Datenfelder oder Attribute, die jeweils durch einen eindeutigen Namen definiert sind. Hier: Kundennummer, Nachname, Vorname, PLZ und Ort.

Jeder Datensatz dieser Tabelle setzt sich zusammen aus genau fünf Datenfeld-Werten und hat somit die gleiche Datenstruktur. Er zeigt zusammengehörige Informationen zu einem Kunden in einer Zeile an. Als Beispiel: Der Kunde 'Michael Müller' mit der Kundennummer '2', Wohnort in 'Köln' und der PLZ = '50939'.

## 1.6 Datensatzaufbau

Der erste Schritt bei der Erstellung einer Datenbankdatei ist die Festlegung des Datensatzaufbaus (Struktur), also die Herstellung der Definition der erforderlichen Datenfelder eines Datensatzes. Für jedes einzelne Datenfeld müssen die folgenden Merkmale angegeben werden:

- Datenfeldname
- Datenfeldtyp
- Datenfeldlänge



**Abd. 1.2 Datensatzaufbau**

### Datenfeld-Name

Der Datenfeld-Name sollte auf den Verwendungszweck des Feldes hinweisen. Die Datenfeldnamen der Tabelle 'Kunde' heißen in diesem Beispiel: 'Kundennummer', 'Name', 'Vorname', 'PLZ' und 'Ort'. Man könnte auch noch als Präfix (Vorsilbe) eine Abkürzung des Datenfeldtyps schreiben. (Siehe „Ungarische Notation“, Seite 31)

### Datenfeld-Typ

Der Datenfeld-Typ richtet sich nach dem geplanten Verwendungszweck. Man unterscheidet hier z. B. Text und Zahlenfelder. Während Textfelder lediglich "starre" Informationen enthalten, können mit den Inhalten der Zahlenfelder Berechnungen durchgeführt werden. Als Beispiel hierfür ist der Datenfeld-Typ für das Datenfeld 'Name' gleich 'Text'.

## Datenfeld-Länge

Für jedes Datenfeld muss eine genaue Länge festgelegt werden, um entsprechend Platz für den Feldeintrag zu schaffen. Z.B. braucht das Feld PLZ genau 5 Zeichen, der Ort braucht sicher mehr und die Kundennummer eventuell auch. Man muss sich also frühzeitig überlegen, wie viel Einträge man an jeder Stelle braucht.

## 1.7 Relationale Datenbank

Microsoft **Access** ist ein *relationales* Datenbankmanagementsystem. Relational ist eine Datenbank, wenn die einzelnen Tabellen untereinander in einer Beziehung (Relation) stehen können. Dieses setzt natürlich voraus, dass die Datenbank aus mehreren Tabellen besteht. Eine nicht-relationale Datenbank kann nur aus einer Tabelle bestehen (bspw. MS Excel). Der Nachteil bei nicht-relationalen Datenbanken sind u. a. redundante, also sich wiederholende Informationen, die man in die Tabelle eintragen muss.

Eine **Access**-Datenbank enthält im Normalfall *mehrere* Tabellen und zusätzlich Informationen darüber, wie die Tabellen zusammenhängen. Ein Beispiel:

An der Universität gibt es eine Datenbank, in der alle Studenten der Universität mit ihren persönlichen Daten (Matrikelnummer, Name, Adresse, Fachbereiche etc.) eingetragen werden. In der Regel ist ein Student in mehreren Fachbereichen eingeschrieben. Bestünde die Datenbank aus nur *einer* Tabelle, müsste für jeden Fachbereich eine eigene Zeile angelegt werden, die jeweils dieselben persönlichen Daten enthält.

*Eine große Tabelle für alle Informationen*

Matrikelnummer	Vorname	Nachname	Adresse	Fachbereich
12345678	Alexandra	von Cube	Dortmund	Mathematik
12345678	Alexandra	von Cube	Dortmund	Kunst
12345678	Alexandra	von Cube	Dortmund	Pädagogik

Erstens enthält diese Tabelle redundante Informationen (Vorname, Nachname, Ort). Zweitens müsste man bei einem Umzug der Person deshalb diesen in jeder Zeile vornehmen. Ein großer Aufwand.

Gibt es nun mehrere Tabellen die man zueinander in Beziehung setzen kann, wird die Änderung der Adresse nur einmal in der Tabelle für Adressen vorgenommen und danach mit der Tabelle für die Fachbereiche mit Hilfe der Matrikelnummer verknüpft. So kann jeder Fachbereich, in den sich der Student eingeschrieben hat, über die Matrikelnummer auf die aktuelle Adresse zugreifen. Das ist mit **Access** möglich.

*Tabelle 1, Adressen (Mastertabelle)*

Matrikelnummer	Vorname	Nachname	Adresse
12345678	Alexandra	von Cube	Dortmund
91011121	Christine	Damberg	Bochum
97654377	David	Niegisch	Tübingen

Tabelle 2, Fachbereiche, die mit der ersten verknüpft ist (Detailtabelle)

Matrikelnummer	Fachbereich
12345678	Mathematik
12345678	Kunst
12345678	Pädagogik
91011121	Hauswirtschaft
91011121	Pädagogik

In diesem Beispiel enthält die erste Tabelle z. B. die persönlichen Daten aller Studenten, die zweite alle Fachbereiche, in denen sich ein Student eingeschrieben hat. Will man nun die Adresse des Studenten herausbekommen, sucht man in der ersten Tabelle nach dem Datensatz, der die jeweilige Matrikelnummer enthält (in *Access* geht das über eine Abfrage sehr schnell, was später erklärt wird).

## 1.8 Was sind Primärschlüsselfelder und Fremdschlüsselfelder?

### 1.8.1 Primärschlüsselfeld

Ein Primärschlüsselfeld ist ein besonderes Feld (Spalte) einer Tabelle. In dieser Spalte darf jeder Wert nur *einmal* vorkommen. Duplikate sind nicht erlaubt. Dieses Feld verknüpft eine Tabelle mit der anderen und dient zur eindeutigen Zuordnung der verschiedenen Daten aus mehreren Tabellen. Ein einzelner Wert in diesem Feld ist der Primärschlüssel.

In der obigen *Tabelle 1* ist die Matrikelnummer der Primärschlüssel. Dieser Wert kommt nur einmal vor, so dass man genau weiß welcher Student gemeint ist, auch wenn Studenten dieselbe Adresse oder denselben Namen haben. Wenn man den Primärschlüssel gestaltet, sollte man ihn sprechend machen, das heißt: Im Primärschlüssel sollte man schon erkennen, was den Datensatz ausmacht. So könnte man eine Personalnummer aus drei Buchstaben Vorname, drei Buchstaben Nachname und einer laufenden Nummer erzeugen. Leider ist das bei den Matrikelnummern nicht der Fall...

Der eigentliche Witz am Primärschlüssel wird deutlich, wenn man zwei Tabellen miteinander verknüpfen, also in Beziehung zueinander setzen will.

### 1.8.2 Fremdschlüsselfeld

Ein Fremdschlüsselfeld ist ein Feld in einer Tabelle, das in einer anderen Tabelle Primärschlüsselfeld ist. Das Fremdschlüsselfeld enthält dieselben Werte, wie das Primärschlüsselfeld, nur dass diesmal die Daten mehrmals, keinmal oder nur einmal vorkommen können. Bezogen auf das obige Tabellenbeispiel bedeutet dies, dass sich ein Student entweder in keinem, einem, oder mehreren Fachbereichen einschreibt.

Der Fremdschlüssel bildet das Gegenstück der Beziehung, verbindet also zwei Datensätze aus unterschiedlichen Tabellen, hier die Datensätze der *Tabelle Adressen* und der *Tabelle Fachbereiche*, verbunden durch das *Feld Matrikelnummer*.

## 1.9 Beziehungen

Dass Tabellen miteinander in Beziehung stehen, wurde schon erwähnt. Eine Beziehung stellt die logische Verbindung zwischen zwei Tabellen dar. Aber wie sehen die Beziehungen genau aus bzw. wie müssen sie aussehen, damit die ganze Datenbank auch funktioniert?



Es gibt drei verschiedene Beziehungstypen zwischen Tabellen in einer Datenbank:

1. die 1 : n-Beziehung
2. die n : m-Beziehung und
3. die 1 : 1-Beziehung.

### 1.9.1 Die 1:n-Beziehung

Die 1er-Seite besteht aus einer Tabelle die keine redundanten Informationen enthält, ein Wert in einer Spalte also nur *einmal* vorkommt. Diese wird auch *Mastertabelle* genannt. Die Tabelle, die die n-Seite der Beziehung darstellt, kann diesen Wert mehrmals ( $n = \text{unendlich}$ ) oder gar nicht enthalten. Diese ist die *Detailtabelle*. Anders gesagt: bei einer derartigen Beziehung entsprechen einem Datensatz (Tabellenzeile) in der Mastertabelle mehrere Datensätze in der Detailtabelle, wobei mehrere auch die Zahlen 0 und 1 einschließt.

#### Beispiel:

Das Prüfungsamt einer Universität braucht eine Datenbank, die in der ersten Tabelle für Matrikelnummern (Mastertabelle) alle Studenten des Fachbereiches Erziehungswissenschaft auflistet. In einer zweiten Tabelle für Prüfungen (Detailtabelle) werden alle Studenten, die in diesem Fachbereich schon eine Prüfung absolviert haben, eingetragen. Weiterhin werden in diese noch Prüfungstitel, Note, Datum und Prüfer vermerkt.

*Mastertabelle (Tabelle Matrikelnummer)*

Matrikelnummer	Nachname	Adressen
12345678	von Cube	Schmitzweg 9
10111213	Müller	Tannenstr. 18
14151617	Schmidt	Harkortstr. 4
18192021	Damberg	Colastr. 8
usw.		

In der Mastertabelle darf die Matrikelnummer nur einmal vorkommen, da das Feld Matrikelnummer Primärschlüssel ist.

*Detailtabelle (Tabelle Prüfungen)*

Matrikelnummer	Prüfung	Note	Datum	Prüfer
12345678	Kolloquium	2,1	13.10. 2001	Meyer
12345678	Zwischenprüfung	2,1	01.02.2003	Schmitz
18192021	Zwischenprüfung	3,0	14.06. 2002	Schmitz
18192021	Kolloquium	1,7	12.05.2000	Meyer
usw.				

In der Detailtabelle wird das Primärschlüsselfeld zum Fremdschlüsselfeld (Matrikelnummerfeld in der Detailtabelle) und stellt die andere Seite der Beziehung dar. Die Logik dieser Tabelle erfordert, dass die Matrikelnummer mehrmals oder gar nicht auftauchen darf. In der Detailtabelle gilt es nämlich festzuhalten, welche Prüfungen Student 1234567 wann und bei wem abgelegt hat. Ein Student legt mit der Zeit mehrere Prüfungen ab, weswegen eine Matri-

kelnummer mehrmals auftauchen können muss. Umgekehrt kann eine Prüfungsbenotung einem Studenten über die Matrikelnummer genau zugeordnet werden. Auch erfährt man, dass Student 14151617 noch keine Prüfungen absolviert hat, da er in der Tabelle für Prüfungen nicht eingetragen ist.

### 1.9.2 Die $n : m$ -Beziehung

In einer  $n : m$ -Beziehung kann ein Datensatz in einem Feld der Tabelle 1 zu einem, keinem oder mehreren Datensätzen in Tabelle 2 in Beziehung gesetzt werden. Umgekehrt besteht von Tabelle 2 die *gleichartige* Beziehung zu Tabelle 1, dass also ein Datensatz der Tabelle 2 zu einem, keinem oder mehreren Datensätzen der Tabelle 1 zugeordnet werden können. In der  $1 : n$ -Beziehung besteht diese Verbindung nur in eine Richtung, von der Mastertabelle zur Detailtabelle.

Die  $n : m$ -Beziehung kann nur indirekt über eine *dritte* Tabelle definiert werden. Die Tabelle 3 enthält die Primärschlüssel der Tabellen 1 und 2 als Fremdschlüssel, wodurch eine genaue Zuordnung der jeweiligen Primärschlüssel definiert ist.

#### Beispiel:

Die Universität will nicht nur die Prüfungen die ein Student abgelegt hat festhalten, sondern auch welcher Student welche Veranstaltungen besucht. Es muss also wieder eine Tabelle geben, in der alle Studenten mit Matrikelnummer z.B. eines Fachbereichs eingetragen sind, eine zweite Tabelle in der alle Seminare mit einer ID und ihren Namen erfasst sind und eine dritte Tabelle, die sowohl Matrikelnummer, als auch die ID-Nummer der Seminare enthält. Diese dritte Tabelle setzt dann Tabelle 1 und 2 in Beziehung.

*Tabelle 1 (Mastertabelle)*

Matrikelnummer	Nachname
12345678	von Cube
10111213	Müller
14151617	Schmidt
18192021	Damberg
usw.	

In Tabelle 1 ist die Matrikelnummer wieder der Primärschlüssel.

*Tabelle 2 (Mastertabelle)*

ID_Seminar	Seminarname	Dozent
0001	Sozioökonomie des Haushalts	Schmetz
0002	Ernährungslehre	Bönnhoff
0003	Wirtschaftslehre	Kuklik
0004	Didaktik der Haushaltslehre	Stiller

In Tabelle 2 ist die ID der Primärschlüssel.

Tabelle 3 (Detailtabelle)

Matrikelnummer	ID_Seminar
12345678	0001
12345678	0002
14151617	0003
12345678	0004
18192021	0003
18192021	0001

In Tabelle 3 sind die Primärschlüssel der Tabellen 1 und 2 zu Fremdschlüsseln geworden, da diese eben mehrmals einmal oder keinmal vorkommen können. Ein Student kann mehrere Seminare besuchen (oder auch keine, was man in diesem Fall nicht hoffen will) und ein Seminar setzt sich wiederum aus mehreren Studenten zusammen. Jeder Student kann jedes Seminar besuchen.

Um diese realen Verhältnisse in Tabellen festhalten zu können, müssen die einzelnen Datensätze der Tabelle 1 und 2 kombinierbar sein. Dies ist in Tabelle 3 der Fall.

### 1.9.3 Die 1 : 1-Beziehung

Der dritte Beziehungstyp tritt relativ selten auf, da die Felder die in der zweiten Tabelle gespeichert werden, eigentlich direkt in der Tabelle 1 gespeichert werden könnten, ohne dass wiederholende Daten auftreten würden. In beiden Tabellen dieser Beziehung existiert der Wert nur einmal und ist in beiden Tabellen Primärschlüssel. Gründe für so eine Beziehung können sein:

- Um eine möglichst schnelle Datenabfrage zu gewährleisten, versucht man die Tabellen klein zu halten. So macht man aus einer Tabelle mit vielen Spalten zwei kleine, deren jeweilige Daten in einer 1 : 1-Beziehung stehen.
- Tabellen können Daten enthalten, auf die Personen zu unterschiedlichen Zwecken zugreifen müssen, aber nicht alle eingetragenen Informationen zu dem Subjekt brauchen oder sehen dürfen.

### 1.10 Referentielle Integrität

Die so genannte referentielle Integrität bestimmt zwei Anforderungen an verknüpfte Tabellen, die für die Richtigkeit der Daten und ihrer Verknüpfung sorgen sollen.

Die beiden Forderungen beinhalten Folgendes:

- Wenn man einen Datensatz in einer *Detailtabelle* einfügt, dürfen nur Fremdschlüssel verwendet werden, die in der *Mastertabelle* vorkommen. In der obigen Tabelle 3 dürfen z.B. nur Studenten und Seminare vorkommen, die es an der Universität auch tatsächlich gibt, also schon in den Mastertabellen 1 und 2 eingetragen sind.
- Man darf nicht einfach einen Datensatz in einer Mastertabelle löschen, wenn dieser mit anderen Detailtabellen verknüpft ist. Dies würde zur Folge haben, dass eine Menge verwaister Datensätze, also ohne Heimat, in Detailtabellen entstehen. Es gäbe Vorlesungen ohne Professoren und Seminare ohne Studenten.

In *Access* lässt sich beim Erstellen der Beziehungen die referentielle Integrität für jede einzelne Beziehung ein- und ausschalten. Das Einschalten empfiehlt sich unbedingt, weil ohne referentielle Integrität das Löschen oder Ändern von Datensätzen, ohne deren komplette Entfer-

nung aus der Datenbank möglich ist. Dies kann zu einer Menge verwaister Datensätze führen was wiederum die Integrität der Daten in der Datenbank zerstört. Ohne Integrität kann man den Daten einer Datenbank jedoch nicht vertrauen. Mit anderen Worten sind ohne den Einsatz von referentieller Integrität die Daten einer relationalen Datenbank ganz schnell nicht mehr vertrauenswürdig und somit eigentlich nur noch Schrott.

## 1.11 Die Normalformen

Normalformen sind Regeln für die Bildung von logischen Datenzusammenhängen, die dann in Tabellen umgesetzt werden.

Wieso ist eine so genannte Normalisierung, also die Anwendung der Normalformen auf die Tabellen, notwendig? Normalisierte Tabellen weisen *erstens* ein Minimum an redundanten Informationen auf, was *zweitens* die physische Größe der Datenbank gering hält. So wird der logische Aufbau klarer, was die Arbeit mit und die Pflege von Datenbanken vereinfacht.

Der Normalisierungsprozess beinhaltet eigentlich 5 Schritte, die streng nacheinander vollzogen werden müssen. In der Praxis verwendet man aber nur die ersten drei, weswegen hier auch nur auf diese eingegangen wird.

Eine Normalisierung beginnt mit der 1. Normalform; ist dieser Schritt abgeschlossen folgt die 2. und dann die 3. Normalform.

Generell kann man sagen, dass ein Normalisierungsschritt mit der Zerlegung einer Tabelle endet, wenn diese die jeweilige Normalform nicht erfüllt hat.

### 1.11.1 Erste Normalform (1. NF)

Eine Tabelle befindet sich in der ersten Normalform, wenn sich die Werte in jeder Zeile und in jeder Spalte im atomaren Zustand befinden, d.h. sich in keine kleineren Einheiten mehr zerlegen lassen.

In der folgenden Tabelle sind alle Informationen zu einem Studenten eingetragen, seien es persönliche Daten, wie Matrikelnummer, Adresse, Name und auch Daten, die das Studium des Studenten betreffen, wie die Fachrichtungen. Weiterhin sind in der Spalte *Adresse* alle Adressanteile, wie Straßename und Stadt untergebracht.

*Tabelle 1: Studenten*

Matrikelnummer	Nachname	Adresse	Geburtsdatum	Fachbereich 1	Fachbereich 2
12345678	von Cube	Schmitzweg 9 44225 Dortmund	20.02. 1980	Musik, EF 50	12
10111213	Schmidt	Tannenstr. 18 45632 Bochum	23.04. 1970	5	13
14151617	Schmidt	Harkortstr. 4 87564 Lünen	15.06. 1982	16	11
18192021	Damberg	Colastr. 8 44123 Dortmund	30.12. 1977	8	5

Es gibt zwei Probleme bei dieser Tabelle:

**Das erste besteht darin**, dass die Werte in der Spalte *Adresse* nicht in atomarer Form vorhanden sind. Die Informationen im Feld *Adresse* sind weiter zerlegbar (atomisierbar), nämlich in Straße, Hausnummer, Ort und Postleitzahl.

Diese Trennung ist notwendig, um bei Abfragen (*siehe Kapitel Abfragen*) möglichst schnell ein Ergebnis zu erzielen. Will die Universität wissen, wie viele Studenten aus welcher Stadt

kommen, würde das Programm bei der obigen Tabelle mehr Zeit für ein Ergebnis brauchen, als wenn die Stadt in einer eigenen Spalte aufgeführt wäre. Zusätzlich ist es wesentlich übersichtlicher, wenn man genau weiß wo der Eintrag Stadt zu finden ist.

Die Spalte *Adresse* muss also durch insgesamt 4 Spalten *Strasse, Hausnummer, Postleitzahl und Ort* ersetzt werden.

**Der zweite Nachteil dieser Tabelle ist**, dass sie nur eine begrenzte Anzahl Spalten für die Fachbereiche hat. Wenn sich nun ein Student in viele Fachbereiche eingeschrieben hat, muss man bei dieser Tabelle Spalten an der rechten Seite anfügen. Dadurch wird die Tabelle ewig breit und manche Felder sind einfach leer, weil sich ja nicht jeder Student in gleich viele Fachbereiche einschreibt.

Wenn man wissen will, welcher Student in welchem Fachbereich studiert, ist eine entsprechende Abfrage bei dieser Tabelle relativ kompliziert und unökonomisch. In welcher Spalte soll man z.B. nach Fachbereich 13 suchen? Dieser Fachbereich kann sich sowohl in der Spalte „Fachbereich 1“ oder „Fachbereich 2“ befinden.

Diese Tabelle muss also sowohl in einer Spalte (Adressen) atomisiert werden, als auch in den Zeilen, damit sie die erste Normalform erfüllt. Dafür braucht man eine zweite Tabelle.

Die folgende Tabelle enthält nur atomare Werte, denn jede logische Einheit (Strasse, Ort, Postleitzahl etc.) hat eine eigene Spalte. Die Abfrage welcher Student in welcher Stadt wohnt ist nun kein Problem mehr.

Tabelle 1: *Persönliche Daten der Studenten (Adresse)*

Matrikelnummer	Nachname	Strasse	Hausnr	Postleitzahl	Ort	Geburtsdatum	Gebäude
12345678	von Cube	Schmitzweg	9	44225	Dortmund	20.02. 1980	1
10111213	Schmidt	Tannenstr.	18	45632	Bochum	23.04. 1970	2
14151617	Schmidt	Harkortstr.	4	87564	Lünen	15.06. 1982	3
18192021	Damberg	Colastr.	8	44123	Dortmund	30.12. 1977	5
usw.							

Die Matrikelnummer ist in Tabelle 1 der Primärschlüssel. Nun fehlt aber noch die zweite Tabelle, die direkte Informationen zum Studium der Studenten, wie die Fachbereiche enthält.

Tabelle 2: *Fachbereiche*

Matrikelnummer	Fachbereich	Abschluss	Semester
12345678	13	Lehramt	2
12345678	14	Lehramt	2
12345678	15	Lehramt	2
14151617	13	Diplom	5
14151617	4	Diplom	4
18192021	15	Lehramt	1

Die *Tabelle 2 Fachbereiche* enthält für jeden Fachbereich, in der sich der Student eingeschrieben hat, eine Zeile. So wächst die Tabelle nicht in die Breite sondern in die Länge. Bei diesem Tabellenaufbau muss die Struktur nicht verändert werden, wenn sich ein Student in mehrere Fachbereiche einschreibt. Die Matrikelnummer ist in dieser Tabelle Fremdschlüssel.

Die Informationen über einen Student sind nun logisch voneinander getrennt. In *Tabelle 1* tauchen nur persönliche Daten der Studenten auf, in der *Tabelle 2* nur Informationen zum

Studium der Studenten. Die Werte in beiden Tabellen sind atomar und erfüllen dadurch die erste Normalform.

### 1.11.2 Zweite Normalform (2. NF):

Eine Tabelle befindet sich in der zweiten Normalform, wenn

- sie sich in der ersten Normalform befindet und
- jede Tabellenzeile nur Informationen enthält, die sich auf das Subjekt (Student) beziehen, das durch den Primärschlüssel (Matrikelnummer) dargestellt wird.

Unsere *Tabelle 1* (persönliche Daten) erfüllt die erste Normalform, jedoch nicht die zweite Bedingung, nämlich dass alle Felder direkt vom Primärschlüssel abhängig sind. Die folgende Tabelle verstößt offensichtlich gegen die zweite Normalform.

Matrikelnummer	Nachname	Strasse	Hausnr.	Postleitzahl	Ort	Geburtsdatum	Gebäude
12345678	von Cube	Schmitzweg	9	44225	Dortmund	20.02. 1980	1
10111213	Schmidt	Tannenstr.	18	45632	Bochum	23.04. 1970	2
14151617	Schmidt	Harkortstr.	4	87564	Lünen	15.06. 1982	3
18192021	Damberg	Colastr.	8	44123	Dortmund	30.12. 1977	5
usw.							

Das Feld Gebäude hat nichts mit der Matrikelnummer zu tun, und gehört in eine andere Tabelle. Dies wäre ein grober Verstoß gegen den logischen Aufbau einer Datenbank, der auch schnell auffallen würde.

### 1.11.3 Dritte Normalform (3. NF)

Die 3. Normalform enthält die Forderungen

- dass die zweite Normalform erfüllt sein muss und
- dass alle Spalten nur einmal vorkommen dürfen und alle Spalten, die nicht den Primärschlüssel bilden, voneinander *unabhängig* sein müssen.

Matrikelnummer	Nachname	Strasse	Hausnr	Postleitzahl	Ort	Geburtsdatum
12345678	von Cube	Schmitzweg	9	44225	Dortmund	20.02. 1980
10111213	Schmidt	Tannenstr.	18	45632	Bochum	23.04. 1970
14151617	Schmidt	Harkortstr.	4	87564	Lünen	15.06. 1982
18192021	Damberg	Colastr.	8	44123	Dortmund	30.12. 1977
usw.						

In dieser Tabelle mit persönlichen Daten kommen zwar alle Spalten nur einmal vor, jedoch ist die Spalte Postleitzahl direkt mit der Spalte Ort verbunden. Sie sind voneinander abhängig. Um der 3. Normalform zu entsprechen muss eine eigene Tabelle nur mit PLZ und Ort erstellt werden (die gibt es auch im Internet) und in die Tabelle mit den Studenten schreibt man nur die PLZ hinein. Der passende Ort wird dann aus der anderen Tabelle mit Hilfe einer Abfrage ermitteln, wenn auf die Daten zugegriffen wird.

Die Tabelle *persönliche Daten*, die alle drei Normalformen erfüllt, sieht dann so aus:

Matrikelnummer	Nachname	Strasse	Hausnr	Postleitzahl	Geburtsdatum
12345678	von Cube	Schmitzweg	9	44225	20.02. 1980
10111213	Schmidt	Tannenstr.	18	45632	23.04. 1970
14151617	Schmidt	Harkortstr.	4	87564	15.06. 1982
18192021	Damberg	Colastr.	8	44123	30.12. 1977
usw.					

## 1.12 Indizes

Indizes sind Zusätze, die man in einer Tabelle definieren kann, um Such- und Sortieroptionen *zu beschleunigen*. Wenn man in einer Tabelle für ein oder mehrere Felder einen Index (Einzahl von Indizes, also dasselbe) definiert, speichert das Datenbanksystem spezielle Informationen über diesen Index in einer getrennten, nicht sichtbaren Systemtabelle.

Mit dem Index kann man z.B. eine bestimmte Reihenfolge der Datensätze definieren, nämlich ab- oder aufsteigend. Ist die Tabelle einmal nach einem bestimmten Kriterium sortiert, merkt sich das *Access*, weswegen man immer schnell auf die Daten zurückgreifen kann. Sucht man einen bestimmten Wert in einer schon sortierten Spalte, läuft dies wesentlich schneller ab, als bei unsortierten Tabellen, weil das Programm nicht die große sichtbare Tabelle sortiert, sondern die unsichtbare, nur zweispaltige Systemtabelle.

Es gibt aber auch Nachteile. Das Suchen und Sortieren läuft zwar schneller, gibt man jedoch neue Daten in die Ausgangstabelle ein, aktualisiert *Access* gleichzeitig die Systemtabelle. Das verzögert den Eingabeprozess. Man sollte also mit Indizes sparsam umgehen und sie nur dort definieren, wo es Sinn macht.

Primärschlüssel und Indizes haben ein festes Verhältnis:

- Ein Primärschlüssel einer Tabelle ist auch immer ein Index. Das wird vom Programm bei der Festlegung des Schlüssels automatisch vorgenommen.
- Umgekehrt muss ein Index nicht Primärschlüssel sein.

Indizes können in mehreren Feldern gesetzt werden, also mehrere Kriterien bilden, nach denen die Daten einer Tabelle sortiert werden sollen. Dafür muss man festlegen, nach welchem Kriterium die Tabelle zuerst sortiert werden soll. Ist die Tabelle nach dem ersten Kriterium sortiert, wird sie nach dem zweiten Kriterium sortiert usw.

### Beispiel:

Ein aufsteigend sortierter Index auf die Felder Nachname, Vorname, Geburtsdatum sortiert nach dem Nachnamen, bei gleichem Nachnamen nach Vornamen, bei gleichem Nach- und Vorname, nach Geburtsdatum.

Die Reihenfolge der Felder wird bei der Definition der Indizes festgelegt.

Indizes können Duplikate, also zweimal denselben Wert in einem Feld, zulassen oder auch nicht. Im letzteren Fall funktioniert er wie ein Primärschlüssel, da dann ein Wert nur einmal eingegeben werden kann.



## 2 Die Benutzeroberfläche von Access 2010

### 2.1 MS Office 2010

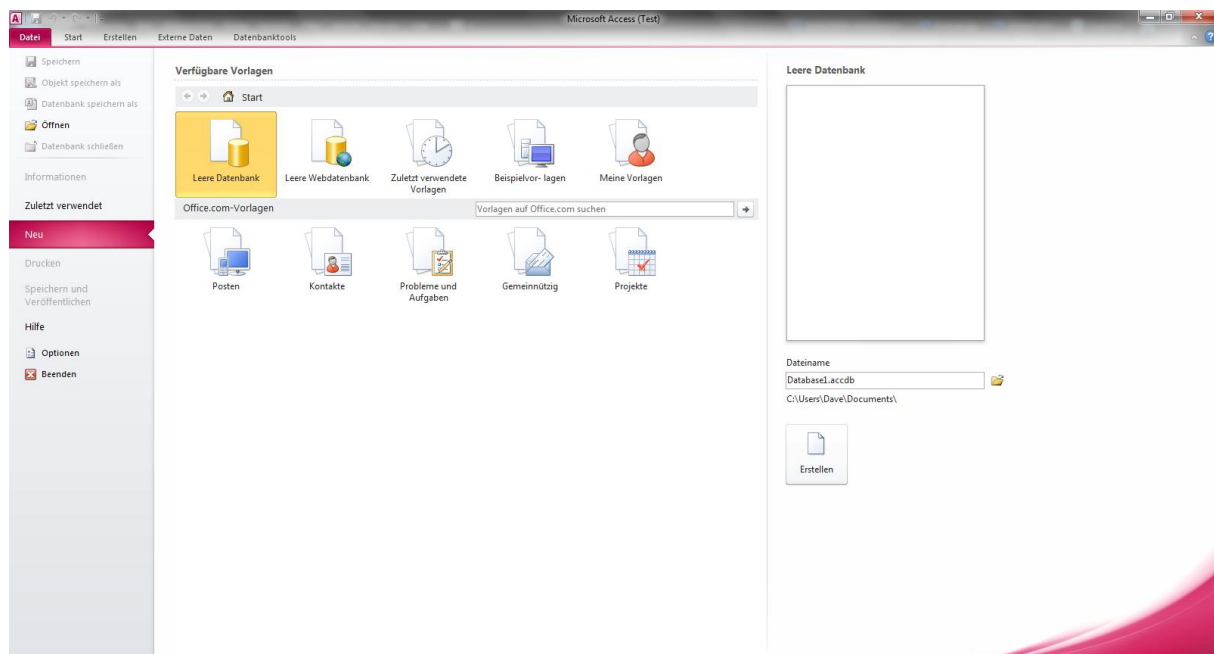
Viele Jahre hat sich am Aussehen und Handling der MS Office-Programme nicht viel geändert. Mit Office 2007 wurde jedoch versucht, einen neuen, benutzer-zentrierten Denkansatz zu realisieren. Der Nutzer und seine Bedürfnisse stehen dabei im Vordergrund und es wurde versucht, aus der Fülle der Programmbefehle eine sinnvolle Vorauswahl zu treffen. Für Einsteiger wurde damit das Erlernen der Access-Grundlagen noch leichter. Mit Office 2010 wurde dieser Ansatz fortgeführt und verfeinert. Die grafische Oberfläche der Programme im Office Paket ist nun noch eine Spur einfacher zu bedienen und intuitiv verständlicher geworden.

Wegen der Änderungen der Benutzungsoberfläche in den verschiedenen Access-Versionen eignet sich dieses Skript ausschließlich für Access 2010. Die Skripte für die Programmversionen 2003 und 2007 finden sich unter folgender Adresse:

<http://yahg.net/kurse/access/>

### 2.2 Die Oberfläche nach dem Start von Access 2010

Nach dem Start erscheint der Eröffnungsbildschirm von *Access 2010*.

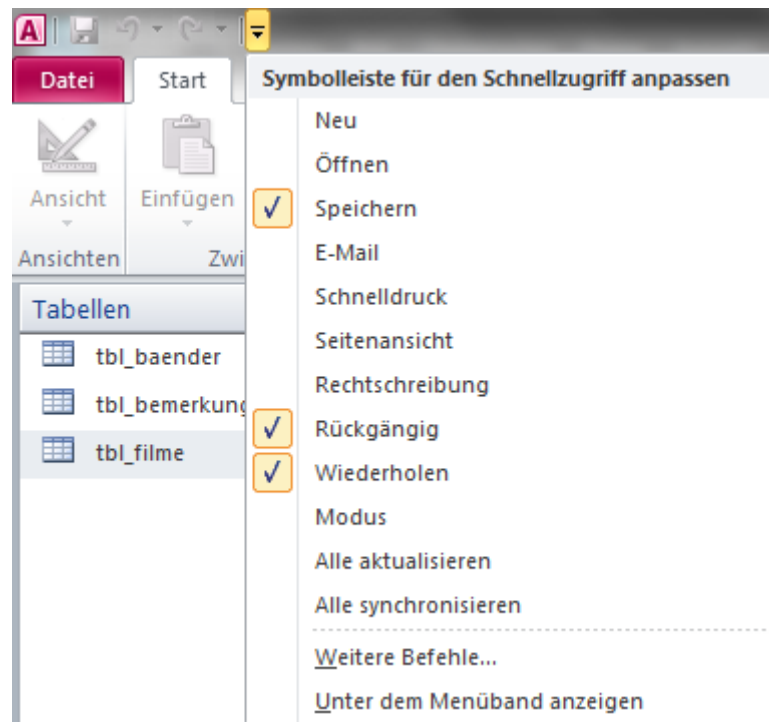


Abd. 2.1 Der Startbildschirm von Access 2010

Eine der Neuerungen seit Office 2007 ist sofort zu erkennen. Die aus älteren Office-Versionen (sowie eigentlich sämtlicher Windows-Software) bekannten Menüleisten mit Befehlen gibt es nicht mehr, sie wurden durch die neue Multifunktionsleiste ersetzt.

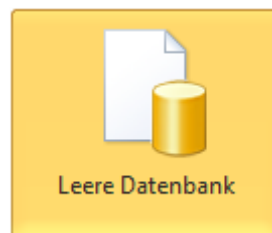
In der obersten Zeile befindet sich die Symbolleiste für den Schnellzugriff, in der sich nach der Neuinstallation lediglich die Icons für die Befehle Speichern, Rückgängig und Wiederholen befinden. Die Symbolleiste für den Schnellzugriff kann vom Nutzer sehr einfach an die eigenen Bedürfnisse angepasst werden.





Abd. 2.2 Symbolleiste für den Schnellzugriff anpassen

Unter **Erste Schritte bei Microsoft Access** *Neue leere Datenbank* *Leere Datenbank* kann man eine neue Datenbank erstellen.



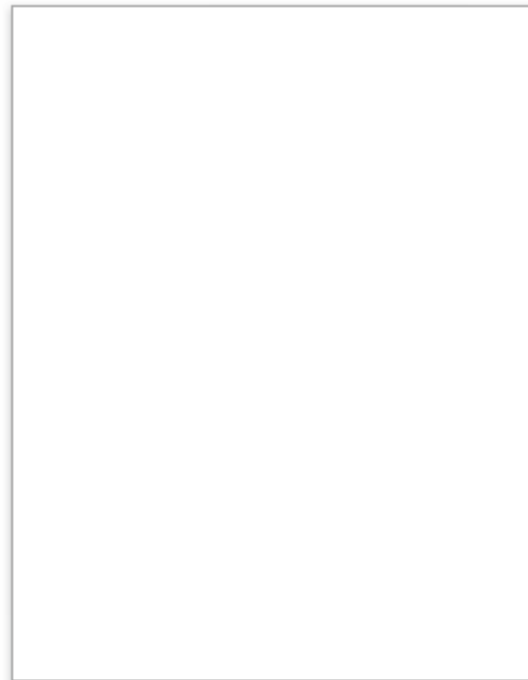
Abd. 2.3 Neue leere Datenbank erstellen

Unter **Zuletzt verwendet** können bereits vorhandene Datenbanken zum Bearbeiten geöffnet werden.

Mit Office 2010 und der Website Office Online steht dem Benutzer nun auch eine Vielzahl von Vorlagen zur Verfügung, die mit wenigen Klicks verwendet werden können. Voraussetzung dafür ist lediglich ein Internetanschluß.

Legt man eine neue, leere Datenbank an, dann passiert etwas, das man von Excel, Word oder Powerpoint nicht kennt. **Access** fordert einen auf, die noch gar nicht erstellte Datenbank zu benennen und zu speichern. Der Dateiname kann in einem Feld auf der rechten Bildschirmseite eingestellt werden.

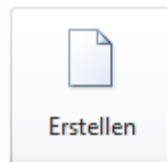
### Leere Datenbank



Dateiname

Database2.accdb 

C:\Users\Dave\Documents\

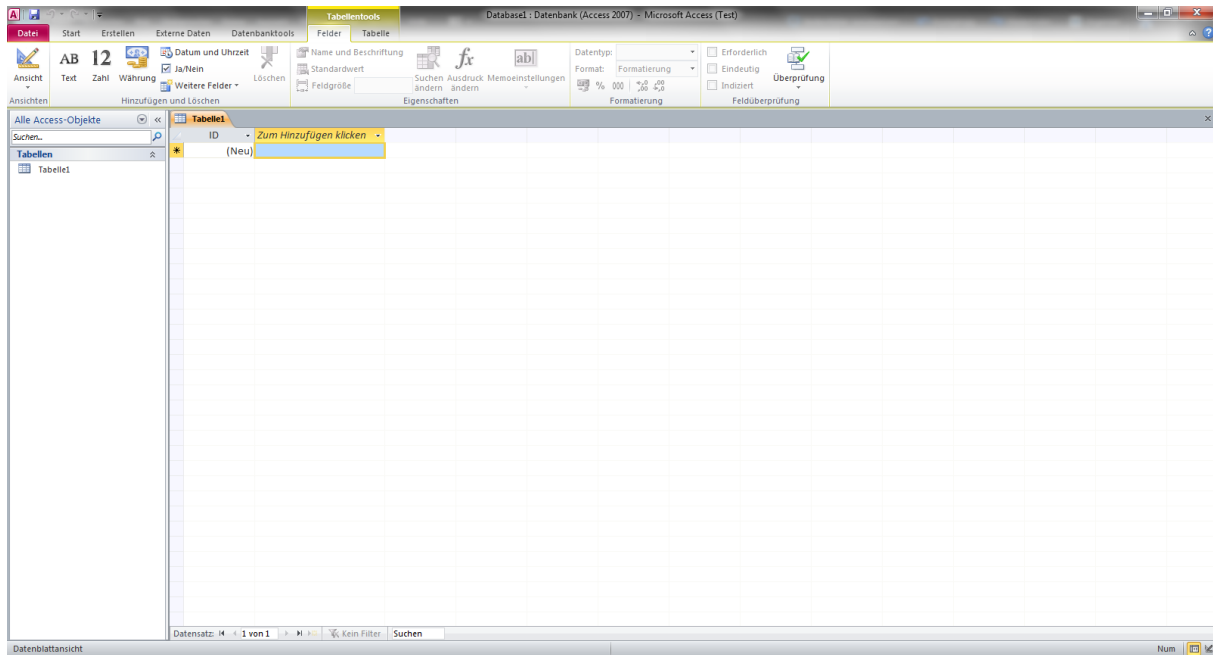


**Abd. 2.4** Hier kann man einen Namen für die neue Datei vergeben

Das liegt daran, dass später bei der Dateneingabe in eine Tabelle nicht mehr separat gespeichert wird. Sobald man einen Datensatz fertig eingegeben hat, wird dieser auf der Festplatte gespeichert. Deswegen braucht *Access* den Dateinamen schon beim Erstellen der neuen Datei.

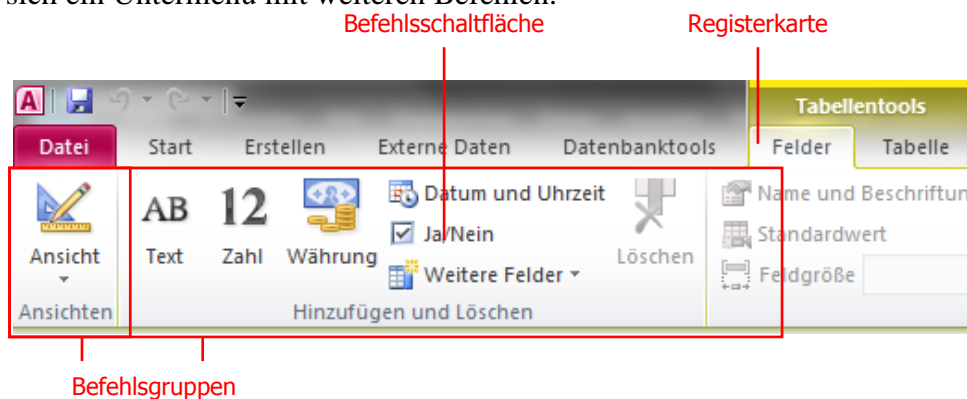
Es ist gut, wenn man sich an dieser Stelle daran gewöhnt, den Dateinamen der Datenbank, in der man seine Tabellen anlegt, entsprechend zu benennen. So kann man sie später leicht von der Datenbank unterscheiden, in der die Aufbereitung der Daten stattfindet. In diesem Skript wird zu diesem Zweck in FrontEnd (FE) und BackEnd (BE) unterschieden. Im BackEnd befinden sich i.d.R. nur Tabellen, während sich Abfragen, Formulare und Berichte im FrontEnd befinden.

Vergibt man einen Namen und klickt auf **Erstellen**, weicht der Eröffnungsbildschirm der Benutzeroberfläche von Access 2010 und man befindet sich automatisch in der **Datenblattansicht** einer neuen Tabelle.



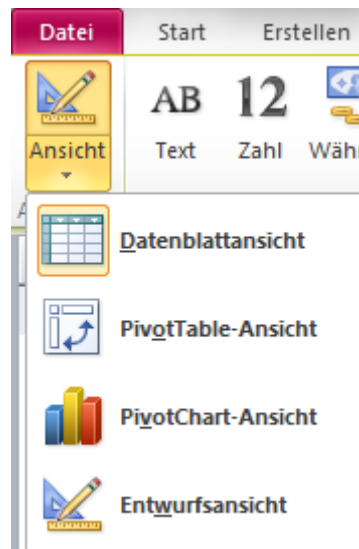
Abd. 2.5 Der look&feel von Access 2010

Ganz oben sieht man die neue Multifunktionsleiste, die die früheren Menü- und Symbolleisten ersetzt hat. Die Multifunktionsleiste setzt sich aus fünf Registerkarten zusammen, zu denen je nach geöffneter Objektart und Ansichtsart noch weitere Register angezeigt werden können. Die Registerkarten enthalten Befehlsgruppen, die Aufgaben in Teilaufgaben unterteilen. So enthält die Registerkarte **Datenblatt** die Befehlsgruppen **Ansichten**, **Hinzufügen** und **Löschen**, **Eigenschaften**, **Formatierung** und **Feldüberprüfung**. Die Befehlsgruppen wiederum enthalten Befehlsschaltflächen, mit denen Befehle direkt ausgeführt werden können oder es öffnet sich ein Untermenü mit weiteren Befehlen.



Abd. 2.6 Die grundlegenden Elemente der Multifunktionsleiste

In der Datenblattansicht der neuen, noch leeren Tabelle kann man sofort anfangen die Tabellen der späteren Datenbank anzulegen. Dies ist eine weitere Neuerung von Access 2007, da man in älteren Programmversionen hierfür in die **Entwurfsansicht** wechseln musste.

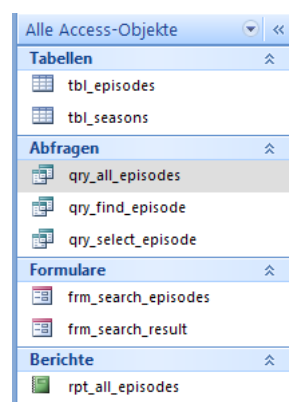


Abd. 2.7 Die Befehlsschaltfläche zum Wechseln der Ansicht

Die Ansicht lässt sich ganz einfach über die entsprechende Schaltfläche in der Multifunktionsleiste ändern.

### 2.3 Was es sonst noch so gibt

Eine Access-Datenbank besteht aus verschiedenen Objekten, die in der Regel im **Navigationsbereich** von Access 2010 angezeigt und ausgewählt werden können. In älteren Programmversionen dient das **Datenbankfenster** einem ähnlichen Zweck.



Abd. 2.8 Die neue Navigationsbereich von Access 2010 ersetzt das Datenbankfenster

Die Objekte haben folgende Funktionen:

- Tabellen** speichern Daten.
- Abfragen** sammeln die angeforderten Daten aus einer oder mehreren Tabellen, damit Sie die Daten in einem Formular ansehen, bearbeiten oder in einem Bericht drucken können.
- Formulare** zeigen Daten aus Tabellen oder Abfragen zusammen an, damit Sie die Daten ansehen, bearbeiten oder eingeben können.
- Berichte** fassen Daten aus Tabellen und Abfragen zusammen und präsentieren sie, damit Sie sie drucken können.

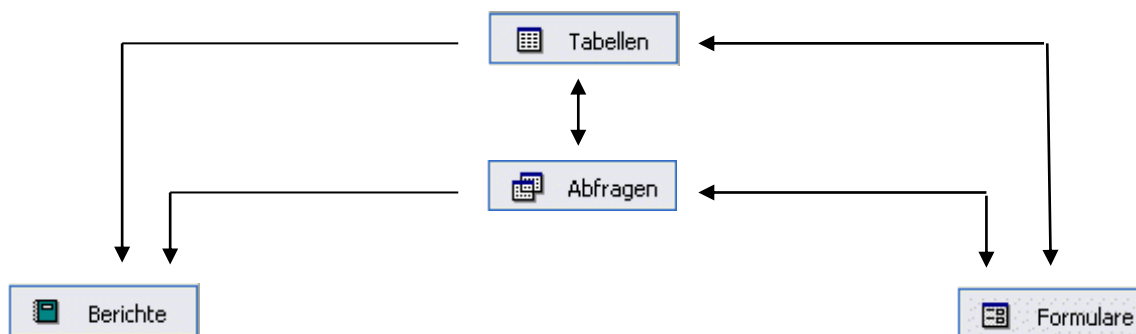
Daneben kann es noch Makros und Module in Access-Datenbanken geben.

**Makros** automatisieren Ihre Datenbank, indem sie ohne Programmierung die von Ihnen angegebenen Aktionen durchführen.

**Module** speichern Visual Basic for Applications (VBA) Code, den Sie schreiben können, um Ihre Datenbank zu erweitern, zu optimieren und ihren Erfordernissen anzupassen.

Makros und Module werden hier nicht weiter erläutert, da dies eindeutig Bereiche für Profis sind. Das sind wir (noch) nicht.

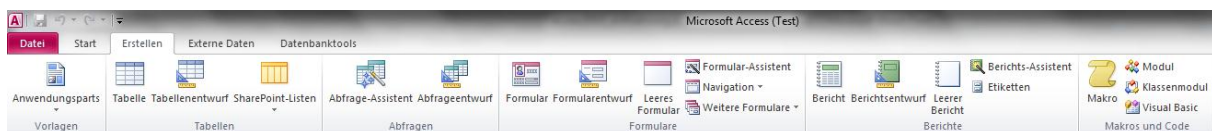
Um die Beziehungen der verschiedenen *Access*-Elemente nachvollziehen zu können, hilft folgende Abbildung:



Abd. 2.9 Beziehung zwischen den Objekten

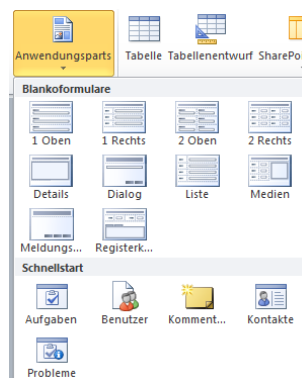
In Worten beschrieben kann man sagen, dass Tabellen die Grundlage einer jeden Access-Datenbank sind. Darauf bauen sämtliche Abfragen, Berichte und Formulare auf, wobei Berichte und Formulare auch auf Abfragen basieren können.

Während in früheren Access-Versionen neue Objekte noch über Schaltflächen im Datenbankfenster erstellt wurden, hat sich dieser Bereich seit *Access 2007* grundlegend verändert. Um in *Access 2010* ein neues Datenbankobjekt anzulegen, muss man den Weg über die Multifunktionsleiste gehen. Die Registerkarte **Erstellen** bietet dabei alle Funktionen zum Erstellen neuer Datenbankobjekte.



Abd. 2.10 Die Registerkarte Erstellen

Neu in Access 2010 ist die Befehlsgruppe **Vorlagen** mit dem Befehl **Anwendungsparts**. Hier finden sich häufig genutzte Vorlagen für bestimmte Aufgaben.



Abd. 2.11 Anwendungsparts

### 3 Die erste Tabelle

#### 3.1 Vorüberlegung

Es sollte bisher klar geworden sein, dass ein Großteil der Arbeit schon vor dem eigentlichen Erstellen der Datenbank in der theoretischen Organisation liegt. Da hilft nur eins: Man muss vorher an den Schreibtisch und mit Papier und Bleistift überlegen, was man eigentlich will. Es ist zwar möglich, Datenbankstrukturen auch im Nachhinein zu ändern, aber das kostet Mühe und ist eventuell auch Quelle von Datenverlusten.

##### 3.1.1 Entity Relationship Model (ERM)

Die übliche Vorgehensweise beim Entwurf einer relationalen Datenbank beinhaltet die folgenden Schritte:

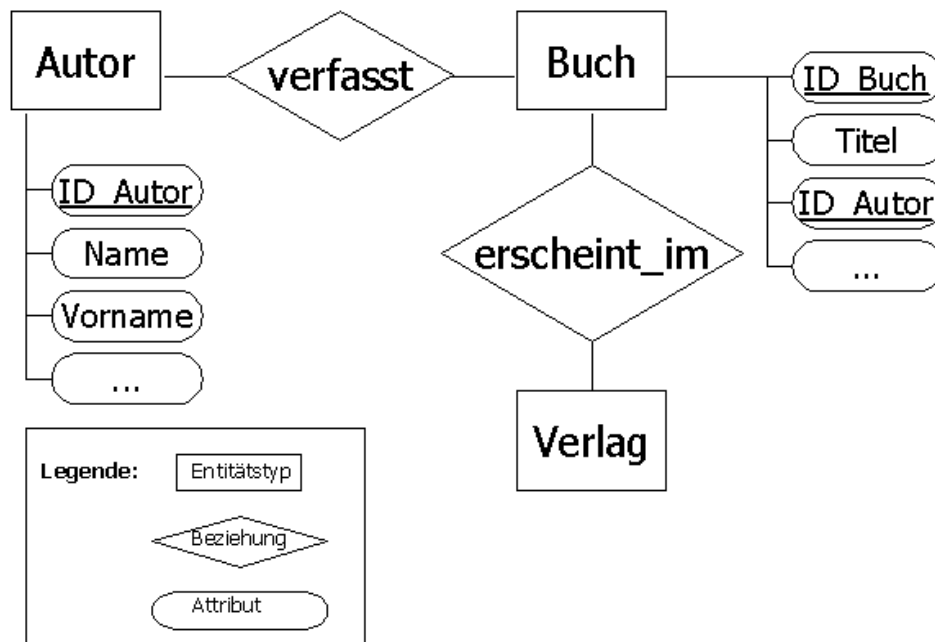
1. Abbildung eines Entity Relationship Model (ERM)
2. Erstellen der Tabellen aus der ERM-Skizze
3. Normalisierung (Anwendung der Normalformen auf die Tabellen)

Das Entity Relationship Model (ERM) ist ein Beschreibungsmittel zur Erstellung von konzeptionellen Schemata. Das ERM beschreibt die permanent gespeicherten Daten und ihre Beziehungen untereinander. Dadurch ist eine Analyse der Informationen aus rein fachlicher Sicht möglich. Die später verwendete Technik, für die Umsetzung des Modells in ein funktionierendes Datenbankmanagementsystem, ist für das ERM nicht von Bedeutung.

Grundlage ist die Typisierung von Objekten und deren Beziehungen untereinander:

- **Entität (Entity):** Identifizierbares Objekt der modellierten Welt (bspw. Autor "Maier", Buch "xyz")
- **Entitätstyp:** Zusammenfassung gleichartiger Entitäten unter einem eindeutigen gemeinsamen Oberbegriff (bspw. Autor, Buch, Verlag, Angestellter, Projekt)
- **Beziehung (Relationship):** semantische Beziehung zwischen zwei oder mehr Entities (bspw. "Autor Maier hat das Buch xyz geschrieben")
- **Beziehungstyp:** Typisierung gleichartiger Beziehungen (bspw. Autorenschaftsbeziehung zwischen Autor und Buch)
- **Attribut:** Elementarinformation einer Entität oder einer Beziehung (bspw. Vorname und Nachname von Autor)

Bei der Datenmodellierung wird in der Regel mit den konkreten Objekten gearbeitet (Entitäten und Beziehungen). Das Entity Relationship Model selbst besteht jedoch immer ausschließlich aus Entitätstypen und Beziehungstypen.



Abd. 3.1 Grafische Darstellung eines Entity Relationship Models (ERM)

### 3.1.2 Aus wie vielen Tabellen soll die Datenbank bestehen?

Die Aufgabe, die in diesem Kapitel zu lösen ist, bezieht sich auf den universitären Bereich.

Die Universität verwaltet Studenten, Professoren, Seminare, Prüfungen etc. mit Hilfe einer Datenbank. An unserer Universität wurde sich für die *Access*-Datenbank entschieden, aber nur weil es im Auenland auf Mitteleerde keine bessere Datenbank gibt und die Anzahl der Studenten 100 nicht übersteigt.

Wenn man weiß, welche Daten in der Datenbank verwaltet werden sollen (hier die oben genannten), ihr den Namen *univerwaltungBE.mdb* gegeben hat, muss man sich die logischen Verbindungen der einzelnen Datengruppen (Matrikelnummer, persönliche Daten der Studenten...) klar machen, um entscheiden zu können, welche zusammen in eine Tabelle gehören und welche nicht. Hierzu dient das Entity Relationship Model (ERM).

Es gibt eine Fülle von Informationen, die sicherlich nicht alle in eine Tabelle gehören, aus den Gründen die im ersten Kapitel genannt worden sind (redundante Informationen, Gewährleistung einer schnellen Abfrage, Übersichtlichkeit). Die Tabellen lassen sich leicht anhand der in Kapitel 1 genannten Normalformen auf diese Dinge hin überprüfen.

Bei unserem Beispiel sind die logischen Zusammenhänge eigentlich klar. Zur Matrikelnummer gehören die persönlichen Daten der Studenten wie der Name, die Adresse usw. und zu den Personalnummern der Professoren natürlich auch deren Namen. Was würde es für einen Sinn machen die Namen der Professoren zusammen mit den Adressen der Studenten in einer Tabelle aufzuführen? Ein großes Durcheinander; die Studenten bekämen die Prüfungsunterlagen ihrer Professoren zugeschickt. Fatal.

Folgende Abbildung zeigt eine Beispieldatenbank für die persönlichen Daten der Studenten. Ähnliche Informationen kann die Tabelle für persönliche Daten der Professoren enthalten.

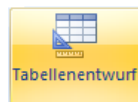
Matrikelnummer	Straße	Hausnummer	PLZ	Wohnort	Geburtsdatum	Körpergröße	Geschlecht	Status	Studiengebühren	Neu
0000001	Kontzestr.	14	22233	DORTMUND	13.04.2004	123 cm	männlich	oS	<input checked="" type="checkbox"/>	
0000002	Singenstr.	123	34565	DORTMUND	12.03.1900	145 cm	weiblich	ZH	<input checked="" type="checkbox"/>	
0000003	Turmstr.	45	85904	MÜNCHEN	13.02.1749	178 cm	männlich	GH	<input type="checkbox"/>	
0000004	Hippenstr.	90	29485	BREMEN	12.09.1790	145 cm	weiblich	GH	<input type="checkbox"/>	
0000005	Teufelsstr.	333	98456	BERLIN	13.04.1979	167 cm	männlich	oS	<input checked="" type="checkbox"/>	
0000006	Gebäudestr.	5	04895	BOCHUM	19.06.1980	187 cm	männlich	oS	<input type="checkbox"/>	
0000007	Schnickstr.	67	83643	FREIBURG	12.03.1974	169 cm	weiblich	GH	<input checked="" type="checkbox"/>	
*	(Neu)					000 cm			<input type="checkbox"/>	

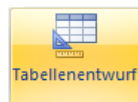
Abd. 3.2 Tabelle für persönliche Daten der Studenten

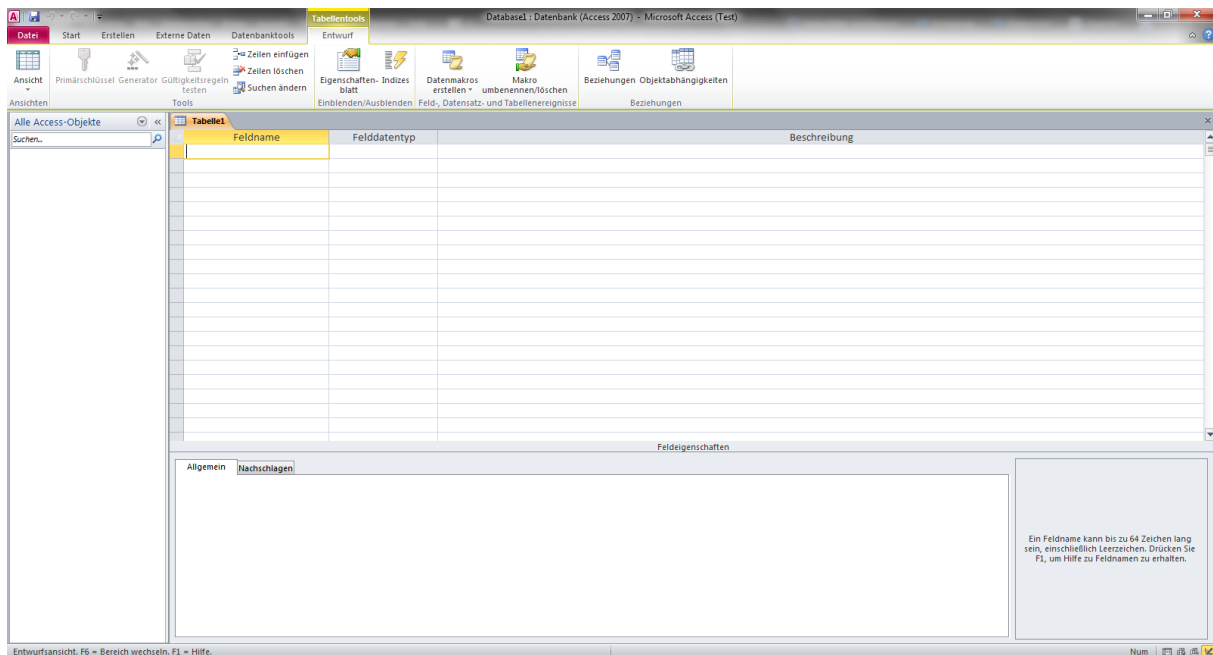
Nun, wie erstellt man so eine schöne Tabelle? In diesem Kapitel soll die Vorgehensweise anhand dieser Beispieltabelle erklärt werden.

In Abd. 3.2 sieht man die Tabelle in der Datenblattansicht, in der man die Informationen über die Studenten eingibt. Seit Access 2007 lassen sich auch in dieser Ansicht neue Felder in einer Tabelle erstellen, was wohl die einfachste Möglichkeit ist. Um jedoch die jeweiligen Feldeigenschaften zu bearbeiten oder gleich bei der Felderstellung richtig anzugeben, muss man in die so genannte Entwurfsansicht wechseln.

### 3.1.3 Die Entwurfsansicht: Feldnamen, Felddatentyp und Feldgröße:



Man öffnet also über den Button  auf der Registerkarte Erstellen in der Multifunktionsleiste eine neue Tabelle in der Entwurfsansicht. Folgendes Bild erscheint:

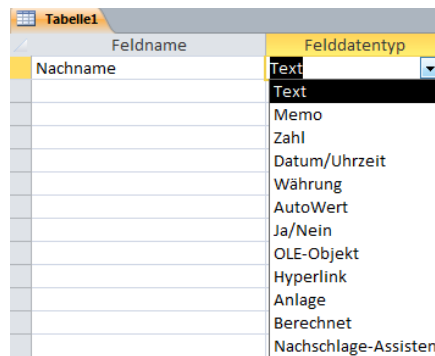


Abd. 3.3 Entwurfsansicht einer Tabelle

In die Spalte ‚Feldname‘ werden die Bezeichnungen für die Tabellenspalten eingegeben. Die Markierung zeigt an, welche Zeile gerade bearbeitet wird.

Der *Felddatentyp* weist den Tabellenspalten verschiedene Eigenschaften zu, je nachdem welcher Datentyp (z.B. Text oder Zahl) in die jeweiligen Spalten eingegeben werden soll. Gibt man einen Feldnamen ein, hier *Nachname*, und klickt in die Spalte Felddatentyp, erscheint ein Button, welcher ein Dropdownfenster öffnet:





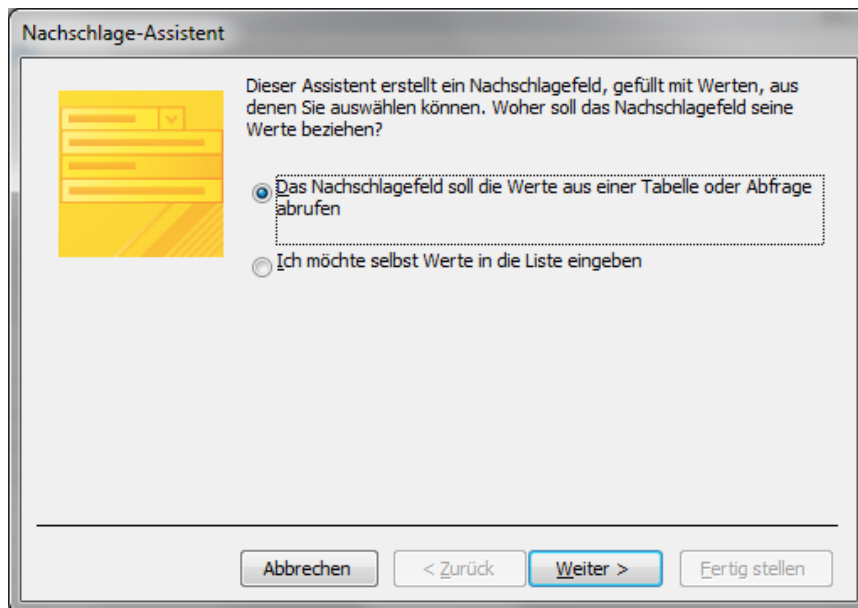
Abd. 3.4 Entwurfsansicht: Felddatentyp

Man sieht, dass es noch mehr Felddatentypen gibt als nur Text oder Zahl, ganz platt gesagt: solche mit denen man rechnen kann, und solche die für Text reserviert sind. In der folgenden Tabelle sind die unterschiedlichen Typen und ihre Funktion aufgelistet.

Felddatentyp	Beschreibung	Größe
Text	Text (Standardeinstellung) erlaubt alle Buchstaben, Zeichen und Zahlen der Tastatur. Allerdings kann man mit den Zahlen nicht rechnen.	Maximal 255 Zeichen, Standardwert sind 50 Zeichen
Memo	Ermöglicht Text von großer Länge, was für Beschreibungen und vertiefende Bemerkungen brauchbar ist.	Maximal 64.000 Zeichen
Zahl	Erlaubt die Eingabe eines beliebigen numerischen Werts. Weitere Präzisionen werden mit der Eigenschaft Feldgröße festgelegt.	1,2,4, oder 8 Byte
Datum/Uhrzeit	Eingabe von Datum- und Zeitangaben	8 Byte
Währung	Eingabe von Datum- und Zeitangaben	8 Byte
AutoWert	Zahl, die von <i>Access</i> automatisch vergeben wird und die eindeutig ist (ohne Duplikat), wenn ein neuer Datensatz einer Tabelle hinzugefügt wird. Felder vom Datentyp <i>AutoWert</i> können nicht verändert werden. Zahlen, die bereits vergeben waren, stehen auch nach dem Löschen der Datensätze nicht mehr zur Verfügung. Man kann bestimmen, ob neue Zahlen jeweils um den Wert 1 erhöht oder als Zufallszahl eingefügt werden. (Siehe Seite 5 - Primärschlüssel)	4 Byte
Ja/Nein	Wahrheitswerte <i>Ja</i> bzw. <i>Nein</i> . Statt <i>Ja</i> ist es erlaubt <i>Wahr</i> oder <i>-1</i> einzugeben. Statt <i>Nein</i> darf man <i>Falsch</i> oder <i>0</i> eingeben.	1 Bit
OLE-Objekt	Objekte die von einem OLE-Server erzeugt wurden (Excel-Tabellen, Draw-Graphik ect.).	128 MB
Hyperlink	Eingabe erscheint als Hyperlink	Bis zu 64.000 Zeichen
Anlage	Neu in Access 2007, ermöglicht das Speichern von Dokumenten und binären Dateien innerhalb der Datenbank.	-
Berechnet	Neu in Access 2010. Ergebnis einer Berechnung. Die Berechnung muss sich auf andere Felder in derselben Tabelle beziehen. Zum Erstellen der Berechnung wird der Ausdrucks-Generator verwendet.	-
Nachschlage-Assistent	Assistent zur Erstellung von Nachschlage-Felder, welche eine Auswahl von Werten aus einer anderen Tabelle/Werteliste ermöglicht.	4 Byte

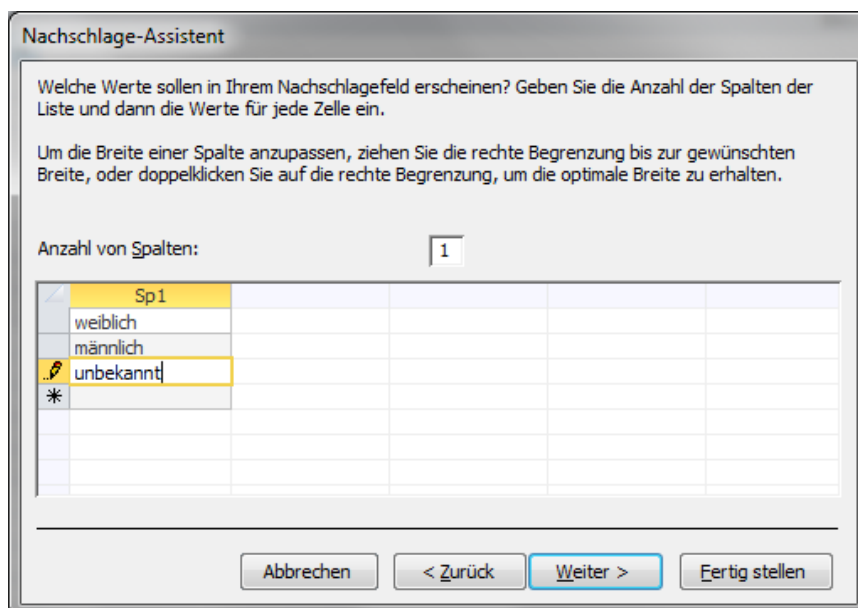
Der **Nachschlage-Assistent** bedarf einer genaueren Erklärung. Eigentlich handelt es sich dabei nicht um einen eigenen Datentyp, sondern um eine Eingabehilfe für Texte, bei der man

eine Werteliste anbietet, aus der man sich einen Eintrag aussuchen kann. Der Assistent führt durch die einzelnen Schritte:



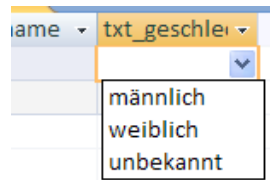
**Abd. 3.5 Erster Schritt beim Nachschlage-Assistent**

Hier bestimmt man, wo die Werteliste herkommt. In diesem Fall, will man sie selber eintragen.



**Abd. 3.6 Zweiter Schritt beim Nachschlage-Assistent**

Im nächsten Schritt trägt man die Werte in die Spalte ein. Und im dritten Schritt gibt man dem Feld einen Namen und zum Schluss hat man ein Dropdownfeld in der Tabelle, aus dem man Werte auswählen kann.



Abd. 3.7 Dropdownfeld zur Auswahl der Daten

Was aber nun, wenn man hier noch einen Wert hinzufügen möchte. Dann muss man wieder in die Entwurfsansicht wechseln, dort das Feld markieren und *unten* in den **Eigenschaften** (ein kleiner Vorgriff auf das nächste Kapitel) die zweite Registerkarte *Nachschlagen* aussuchen.

Allgemein	Nachschlagen
Steuerelement anzeigen	Kombinationsfeld
Herkunftstyp	Wertliste
Datensatzherkunft	"männlich";"weiblich";"unbekannt"
Gebundene Spalte	1
Spaltenanzahl	1
Spaltenüberschriften	Nein
Spaltenbreiten	2,54cm
Zeilenanzahl	16
Listenbreite	2,54cm
Nur Listeneinträge	Nein
Mehrere Werte zulassen	Nein
Wertlistenbearbeitung zu	Ja
Bearbeitungsformular für	
Nur Datensatzherkunftsw	Nein

Abd. 3.8 Die Werteliste des Dropdownfelds

Hier findet man die Werteliste durch Semikolons getrennt und fügt den gewünschten Eintrag einfach hinzu.

Zurück zum oberen Teil der Eigenschaften, wo man als letztes die Spalte "*Beschreibung*" findet. Diese sollte man ausfüllen, wenn der Feldname nicht eindeutig ist. Zum einen kann man hier auf Besonderheiten aufmerksam machen, z. B. ob es Gültigkeitsregeln für die Eingabe gibt. Zum anderen weiß man dann auch nach drei Wochen noch, was man mit diesem Feld eigentlich anfangen wollte.

### 3.1.4 Feldeigenschaften

Im unteren Teil der Entwurfsansicht kann man die verschiedenen **Feldeigenschaften** bestimmen. Welche *Feldeigenschaften* es gibt, hängt davon ab welcher *Felddatentyp* (Text, Zahl, AutoWert etc.) definiert worden ist.

Allgemein	Nachschlagen
Feldgröße	255
Format	
Eingabeformat	
Beschriftung	
Standardwert	
Gültigkeitsregel	
Gültigkeitsmeldung	
Eingabe erforderlich	Nein
Leere Zeichenfolge	Ja
Indiziert	Nein
Unicode-Kompression	Ja
IME-Modus	Keine Kontrolle
IME-Satzmodus	Keine
Smarttags	

Abd. 3.9 Feldeigenschaften beim Textfeld

Im Folgenden werden die Feldeigenschaften für den Datentyp **Text** erläutert.

Die **Feldgröße** bestimmt, wie viel Speicherplatz das jeweilige Feld *maximal* für die Daten zur Verfügung stellt.

Bei einem Textfeld gibt man die benötigte Zeichenanzahl (0-255) einfach per Hand ein. Der Standardwert, also die Voreinstellung, beträgt 255 Zeichen.

Damit kann ein wenig das Eingabeverhalten gesteuert werden. Wenn man weiß, dass der Eintrag niemals 5 Zeichen übersteigt (z.B. bei der Postleitzahl), kann man über die Eingabegrenzung Fehler vermeiden.

Mit der Eigenschaft **Format** kann man bestimmen, wie die eingegebenen Daten angezeigt werden sollen. Es gibt vordefinierte Formate, die **Access** in einem Dropdownfenster anbietet. Man kann auch benutzerdefinierte Formate über die Eingabe von Zeichen in die Zeile *Format* erstellen. Eine Auswahl der zur Verfügung stehenden Zeichen und ihrer Funktionen sind in den folgenden Tabellen aufgeführt, wobei das bei dem Textfeld nicht ganz so wichtig ist, da man im Normalfall wohl genau das sehen möchte, was man dort hineinschreibt. Am Wichtigsten ist hier wohl die Möglichkeit, die Groß- und Kleinschreibung zu erzwingen:

Symbole für *alle* Datentypen

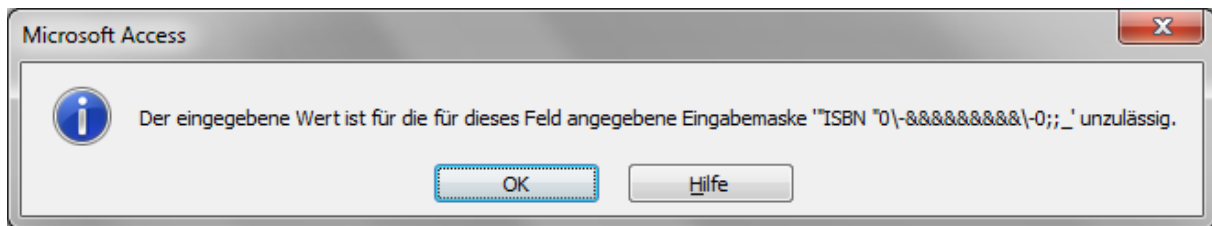
Symbol	Bedeutung
(Leerzeichen)	Zeigt ein Leerzeichen an.
"ABC"	Eine beliebige Zeichenfolge in Anführungszeichen wird unverändert wiedergegeben.
!	Ausrichtung ist linksbündig statt rechtsbündig.
*	Füllt verfügbaren Leerraum mit dem auf das Sternchen nachfolgende Zeichen auf.
\	Nur das dem Backslash folgende Zeichen wird unverändert angezeigt.
[Farbe]	Zeigt die formatierten Daten in der in eckigen Klammern angegebenen Farbe an. Verfügbare Farben: Schwarz, Blau, Grün, Zyan, Rot, Magenta, Gelb, Weiß.

Besondere Symbole für Datentypen *Text* und *Memo*

Symbol	Beschreibung
<	Alle Zeichen in Kleinbuchstaben.
>	Alle Zeichen in Großbuchstaben.

Eine ausführliche Auflistung vorhandener Zeichen und ihrer Wirkung auf das Erscheinungsbild der Daten für die anderen Datentypen kann man der Hilfe von **Access** entnehmen. Die wird über die Taste **F1** geöffnet, wobei der Cursor in der Zeile *Format* stehen muss.

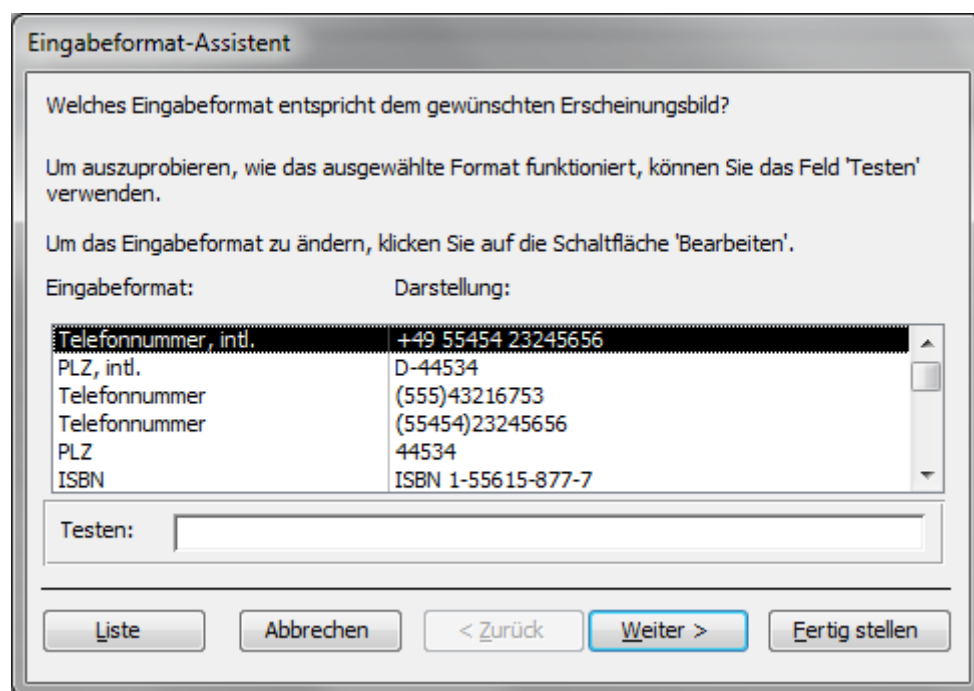
Im Gegensatz zu der Feldeigenschaft *Format*, welche einzig und allein das Datenanzeigeformat nach der Eingabe festlegt, kann man über die Feldeigenschaft **Eingabeformat** die Dateneingabe an sich beeinflussen und vereinfachen. Man hat die Möglichkeit, zu bestimmen, wo und in welcher Anzahl Daten in ein Feld eingegeben werden dürfen bzw. müssen. Falls die Eingabe nicht dem definierten Eingabeformat entspricht, erscheint beim Verlassen des Feldes die in Abd. 3.10 zu sehende Fehlermeldung.



Abd. 3.10 Fehlermeldung am Beispiel einer ISBN-Maske

Eingabeformate können für die Felder der Datentypen Text, Zahl oder Währung und Datum/Uhrzeit festgelegt werden. Das ist insbesondere dann sinnvoll, wenn die Daten eines Feldes immer denselben Aufbau besitzen, wie zum Beispiel eine ISBN oder Matrikelnummer.

Eingabeformate können sehr einfach über den Eingabeformat-Assistenten (Abd. 3.11) festgelegt werden.



Abd. 3.11 Eingabeformat-Assistent

Hat man allerdings vor, ein eigenes Eingabeformat zu erstellen, muss man die entsprechenden Platzhalter kennen:

Zeichen	Beschreibung
0	Ziffer (0 bis 9, Eingabe erforderlich, Plus- [+] und Minuszeichen [-] sind nicht erlaubt).
9	Ziffer oder Leerzeichen (Eingabe nicht erforderlich, Plus- und Minuszeichen sind nicht erlaubt).
#	Ziffer oder Leerzeichen (Eingabe nicht erforderlich, Leerzeichen werden als Leerzeichen im Bearbeitungsmodus angezeigt, aber beim Speichern der Daten entfernt, Plus- und Minuszeichen sind erlaubt).
L	Buchstabe (A bis Z, Eingabe erforderlich).
?	Buchstabe (A bis Z, Eingabe optional).
A	Buchstabe oder Ziffer (Eingabe erforderlich).
a	Buchstabe oder Ziffer (Eingabe nicht erforderlich).
&	Ein beliebiges Zeichen oder ein Leerzeichen (Eingabe erforderlich).
C	Ein beliebiges Zeichen oder ein Leerzeichen (Eingabe nicht erforderlich).
. , ; - /	Platzhalter für Dezimaltrennzeichen sowie Tausender-, Datums- und Zeit-Trennzeichen. (Das tatsächlich verwendete Zeichen hängt von den Einstellungen im Dialogfeld <b>Eigenschaften von Ländereinstellungen</b> in der Systemsteuerung von Windows ab.)
<	Alle Buchstaben werden in Kleinbuchstaben umgewandelt.

Zeichen	Beschreibung
>	Alle Buchstaben werden in Großbuchstaben umgewandelt.
!	Bewirkt, dass die Anzeige im Eingabeformat von rechts nach links anstatt von links nach rechts erfolgt. Eingegebene Zeichen füllen das Eingabeformat immer von links nach rechts aus. Sie können das Ausrufezeichen-Symbol an jeder beliebigen Stelle im Eingabeformat einfügen.
\	Das folgende Zeichen wird als Literal angezeigt, die Wirkung als Sonderzeichen wird ggf. dadurch aufgehoben. (z. B. wird \A als A angezeigt).

An Hand von ein paar Beispielen wird es noch klarer:

Eingabeformat	Beispielwerte
(00000) 000-0000	(0231) 721-4642
(999) 999-9999	(0231) 721-4642
	( ) 555-0248
(00000) AAA-AAAA	(02206) 555-TELE
#999	-20
	2000
>L???L?000L0	GRÜN GR339M3
	MAI R 452B7
>L0L 0L0	T2F 8M4
00000-9999	98115-
	98115 -3007
>L<????????????	Maria
	Kruggel
ISBN 0-&&&&&&&&-0	ISBN 1-55615-507-7
>LL00000-0000	DB51392-0493

Mit **Beschriftung** weist man den Tabellenspalten ihre Überschriften zu, die man in der Datenblattansicht auch sehen kann. Die Feldnamen in der Entwurfsansicht dienen mit ihren Präfixen nur der Orientierung innerhalb der Datenbank. Wenn das Feld nicht ausgefüllt wird, setzt **Access** automatisch den schon vergebenen Feldnamen ein.

Das Feld **Standardwert** erlaubt eine Voreinstellung des Wertes. Wenn man zum Beispiel weiß, dass die meisten Studenten der Universität Dortmund auch in Dortmund wohnen, kann man als Standardwert in der Spalte *Wohnort* „Dortmund“ angeben. Es ist kein Problem diesen Wert dann wieder in der Datenbank zu ändern.

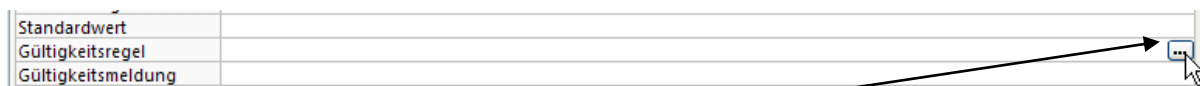
**Gültigkeitsregel:** Man kann über so genannte Ausdrücke, Gültigkeitsregeln für die jeweiligen Felder formulieren. Wird eine Regel für ein Feld festgelegt, können nur solche Werte eingegeben werden, die der Gültigkeitsregel entsprechen. Eine Gültigkeitsregel schränkt den bereits durch den Felddatentyp vorgegebenen Rahmen der Dateneingabe weiter ein.

Beispiele für Gültigkeitsregeln sind in der folgenden Tabelle aufgelistet:

Gültigkeitsregel	Bedeutung
$\geq 0$	Wert muss größer oder gleich Null sein
$>12$ UND $\leq 44$	Wert muss zwischen 12 und 44 liegen, die obere Grenze mit eingeschlossen
$>$ Datum	Datum muss mindestens einen Tag nach dem aktuellen Tagesdatum liegen
$\geq$ Datum()-Tag(Datum()+1	Datum muss nach dem 1. des laufenden Monats liegen oder diesem gleich sein
$\geq$ DatSeriell(Jahr(Datum());1;1) Und $<$ DatSeriell(Jahr(Datum()+1);1;1)	Datum muss ein Wert des aktuellen Jahres sein
Wie "M*"	Der Text muss mit einem M beginnen. Art und Menge der folgenden Zeichen sind beliebig
In(„Jan“;„Feb“)	Erlaubt nur die Eingabe des in Anführungszeichen stehenden Textes (hier: ‚Jan‘ und ‚Feb‘)

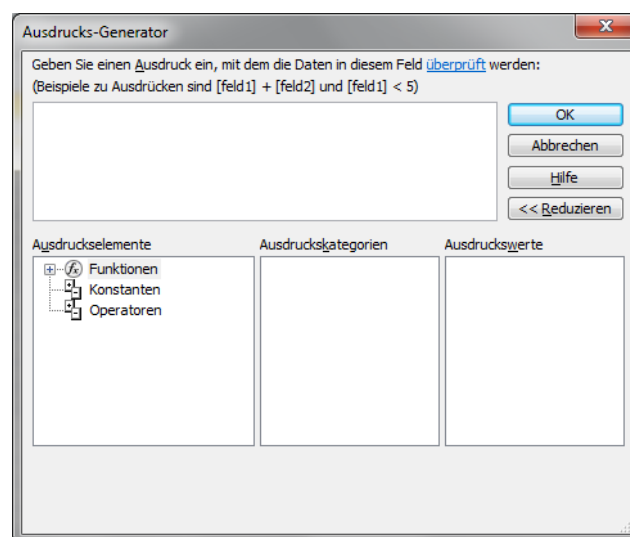
### Kleiner Exkurs am Rande:

Man kann Ausdrücke per Hand eingeben, aber auch mit Hilfe des so genannten Ausdrucks-Generators. Klickt man in die Feldeigenschaft **Gültigkeitsregel** kann man über den auf der rechten Seite erscheinenden Knopf den Ausdrucks-Generator öffnen.



Abd. 3.12 Knopf für Ausdrucks-Generator

Nun erscheint folgendes Fenster:



Abd. 3.13 Ausdrucks-Generator

Der Ausdrucks-Generator kann immer dann geöffnet werden, wenn **Access** das Erstellen von Ausdrücken ermöglicht. Bei kurzen Ausdrücken, wie zum Beispiel =Datum(), welcher in jedes Feld das aktuelle Datum einfügt, wird man den Generator nicht brauchen. Werden die Ausdrücke jedoch länger, wie in der vorherigen Tabelle zu sehen, kann er eine große Hilfe sein. Auf die Ausdrücke wird in späteren Kapiteln noch genauer eingegangen.



Nun weiter bei den Feldeigenschaften.

Eine **Gültigkeitsmeldung** ist wichtig, um bei einer Falscheingabe zu erfahren, was denn falsch war bei der Eingabe. Besonders wenn noch andere mit der Datenbank arbeiten, ist diese Meldung unerlässlich. Den Text, den man in der Fehlermeldung sehen will, gibt man ohne Anführungszeichen in die Zeile ein.

**Eingabe erforderlich** bestimmt, wie es schon sagt, ob man eine Eingabe machen muss oder nicht. Im Fachjargon spricht man dann vom Zulassen von Nullwerten (Eingabe erforderlich =Nein) oder nicht (Eingabe erforderlich =Ja).

**Leere Zeichenfolge** bestimmt, ob eine leere Zeichenfolge (doppelte Anführungszeichen hintereinander: "") erlaubt sein soll oder nicht.

Von „leerer Zeichenfolge“ sind „Nullwerte“ zu unterscheiden. Mit Nullwert ist gemeint, dass das Feld komplett leer gelassen wird.

Mit der Kombination der beiden Feldeigenschaften „Leere Zeichenfolge“ und „Eingabe erforderlich“ kann man unterschiedliche Eingabevorgaben erwirken. Ein Beispiel: Wenn man sowohl Nullwerte wie auch leere Zeichenfolge zulässt (Eingabe erforderlich = Nein, leere Zeichenfolge = Ja), kann man zwischen zwei leeren Werten unterscheiden. Dies kann beispielsweise sinnvoll sein, um zwischen unbekanntenen Werten und (noch) nicht verfügbaren Werten zu unterscheiden.

Andererseits ist es im Feld Matrikelnummer sinnig, durch *Eingabe erforderlich =Ja* die Eingabe zu erzwingen. Nur so kann jeder Student eindeutig identifiziert werden.

**Indiziert** gibt an, ob in dem Feld ein Index gesetzt worden ist oder nicht (*siehe Kapitel 1.6 Indizes*). Der im Dropdownfenster erscheinende Befehl (*siehe Abd. 3.14*) „Ja (Ohne Duplikate)“ verhindert doppelte Werte, der Befehl „Ja (Duplikate möglich)“ erlaubt eine Eingabe desselben Wertes in diesem Feld.

Indiziert	Nein
Unicode-Kompression	Nein
IME-Modus	Ja (Duplikate möglich)
IME-Satzmodus	Ja (Ohne Duplikate)

Abd. 3.14 Feldeigenschaft: Indiziert

Die restlichen Feldeigenschaften sind nur bei einer Programmierung wichtig, was in diesem Kurs nicht interessiert.

Die Feldeigenschaften, die bei **Zahlenfeldern** abweichen sind folgende:

Bei einem Zahlenfeld gibt es die in der folgenden Abd. 3.15 aufgelisteten *Feldgrößen* (die erscheinen automatisch in einem Dropdownfenster, wenn man den Cursor in der Spalte Feldgröße stehen hat):

Allgemein	Nachschlagen
Feldgröße	Long Integer
Format	Byte
Dezimalstellenanzeige	Integer
Eingabeformat	Long Integer
Beschriftung	Single
Standardwert	Double
Gültigkeitsregel	Replikations-ID
Gültigkeitsmeldung	Dezimal

Abd. 3.15 Feldgröße

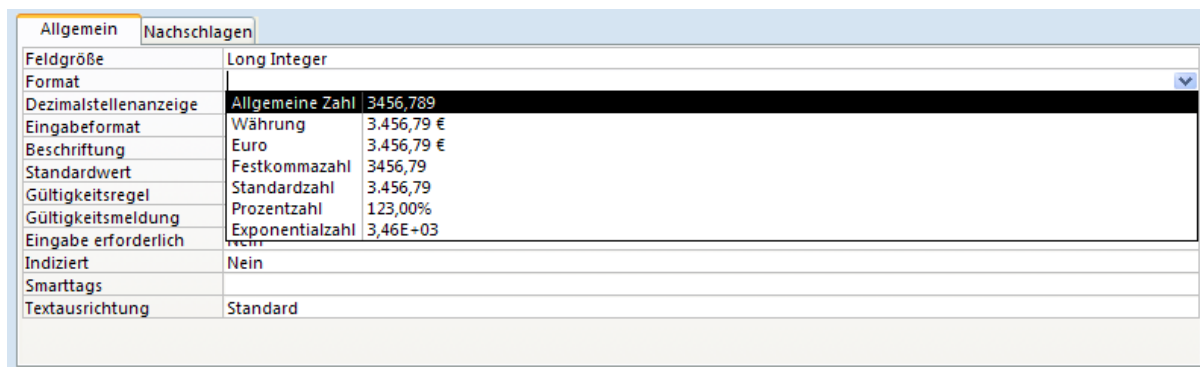
Diese Größen haben folgende Bedeutungen:



Feldgröße	Zahlenbereich	Dezimalstellen	Speicherplatz
Byte	0 bis 255	keine	1 Byte
Integer	-32 768 bis 32 767	keine	2 Byte
Long Integer	-2 147 483 648 bis 2 147 483 647	keine	4 Byte
Single	-3,402823E38 bis -1,401298E-45 für negative Werte; 1,401298E-45 bis 3,402823E38 für positive Werte	7	4 Byte
Double	-1,79769313486232E308 bis -4,94065645841247E-324 für negative Werte; 4,94065645841247E-324 bis 1,79769313486232E308 für positive Werte	15	8 Byte

Integer bedeutet ganze Zahlen, also sind keine Dezimalstellen erlaubt.

Auch für das Zahlenfeld gibt es *Formate*, die man aus einem Dropdownfenster auswählen kann.



Abd. 3.16 Format bei Zahlenfeldern

Aber auch hier kann man sich selber Formate basteln. Dies mit folgenden Symbolen:

#	zeigt nur signifikante Ziffern an, nichtsignifikante Nullen werden ignoriert. Sind bei den Nachkommastellen weniger # angegeben als benötigt, wird gerundet.
0	(Null) zeigt nichtsignifikante Nullen an, wenn eine Zahl weniger Stellen aufweist als Nullen im Format vorhanden sind.
?	fügt auf beiden Seiten der Dezimalstelle Leerzeichen für nichtsignifikante Nullen ein, um Dezimalzahlen am Dezimalkomma auszurichten, wenn die Formatierung mit einer Festbreitenschrift erfolgt (beispielsweise Courier New). Man kann das Zeichen ? auch für Brüche mit einer unterschiedlichen Anzahl von Ziffern verwenden. Allerdings ist es als Formatierungszeichen doch relativ selten.
# „xxx“	Hängt an die eingegebene Zahl einen Text an. Das ist für das Erstellen von Einheiten wichtig, da man in ein Zahlenfeld ja keinen Text schreiben darf.

*Signifikante* Ziffern sind für die Darstellung einer Menge *bedeutsame* Ziffern.

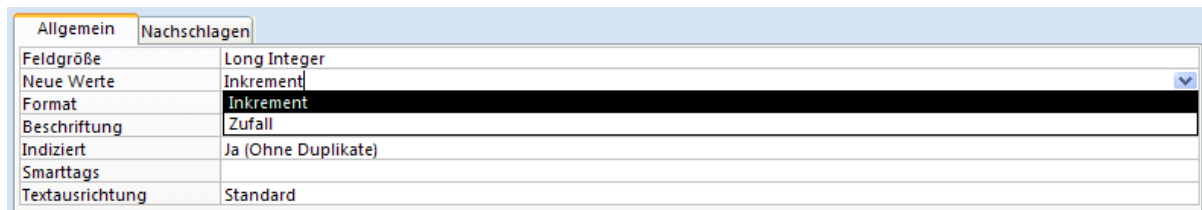
Bei der Zahl 3,9800 sind die letzten beiden Nullen unwichtig, weil sie nichts weiter zur Darstellung der Menge beitragen.

Hier ein paar Beispiele:

Zahlenformat	Code
1234,59 als 1234,6	####,#
8,9 als 8,900	#,000
,631 als 0,6	0,##
12 als 12,0 und 1234,568 als 1234,57	#,0##
44,398, 102,65 und 2,8 mit ausgerichteten Dezimalstellen	???,???
120 m <sup>2</sup>	#" m <sup>2</sup> "

Die Feldeigenschaft *Dezimalstellenanzeige* macht das, was es vorgibt zu tun. Es bestimmt die Anzahl der Stellen, die nach dem Komma folgen sollen.

Die Feldeigenschaft, die beim **Datentyp Auto-Wert** hinzukommt ist *Neue Werte*. Sie bestimmt, wie neu hinzukommende Daten nummeriert werden sollen. *Inkrement* bewirkt, dass automatisch ein Zähler höher gezählt wird, bei *Zufall* werden eben zufällig Zahlen vergeben.



Abd. 3.17 Feldeigenschaft *Neue Werte*

## 3.2 Ungarische Notation

Bevor man dann wirklich zur Tat schreitet, noch eine wichtige Regel, an die man sich auch halten sollte.

Um innerhalb einer Datenbank die unterschiedlichen Objekte als Tabelle, Abfrage oder Bericht identifizieren zu können, gibt es Typkürzel, die an den Anfang des Objektnamens gesetzt werden. Das ist in Datenbanken Standard, so dass jeder Fremde, der mit einer Datenbank zu tun bekommt, gleich erkennen kann, hinter welcher Datei sich welches Objekt verbirgt. Diese Typkürzel werden „Ungarische Notation“ genannt, einfach deswegen, weil der Erfinder aus Ungarn stammt. Nur Folgende Typkürzel sind in diesem Skript allerdings wichtig:

Objekttyp	Präfix
Tabelle (Table)	tbl
Abfrage (Query)	qry
Bericht (Report)	rpt
Formular (Form)	frm

Googelt man im Internet, findet man weitere Informationen dazu.

Wenn man ganz genau arbeiten möchte, sollte man auch die Feldnamen mit den richtigen Präfixen versehen. Dann kann man schon am Feldnamen erkennen, um welchen Felddatentyp es sich handelt. Folgende Abkürzungen sind hier wichtig:

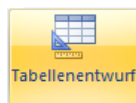
Felddatentyp		Präfix
Text		txt
Datum/Uhrzeit		dat
Ja/Nein		bool
Währung		cur (currency)
Zahlentyp		
	Integer	int
	Long	lng
	Single	sng
	Double	dbl
	Byte	byte

Primärschlüssel werden in diesem Skript immer mit „ID“ als Präfix versehen. Umlaute und Leertasten sollte man generell vermeiden, damit es keine Probleme mit anderen Systemen gibt.

### 3.3 Eingabe von Feldnamen und ihren Eigenschaften

Die Zuordnung unserer Spaltenüberschriften aus dem ersten Kapitel wollen wir mit einer Übung einmal durchexerzieren.

#### Übung



Öffnen Sie über den Button **Tabellendesign** der sich in der Multifunktionsleiste auf der Registerkarte **Erstellen** befindet, die Entwurfsansicht einer neuen Tabelle. Wenn sie schon eine geöffnet haben, umso besser.

- Wir haben die Entwurfsansicht einer neuen Tabelle vor uns. Der Cursor blinkt in der ersten Zeile der Spalte *Feldname*.

Feldname	Felddatentyp	Beschreibung

Abd. 3.18 Tabelle 1 in der Entwurfsansicht

Wir geben den ersten Feldnamen (Spaltenüberschrift) unserer Tabelle ein:

ID\_matrikel

- Sobald Sie das fertig eingegeben haben und auf die TAB-TASTE drücken, springt der Cursor in die Spalte *Felddatentyp*, TEXT wird automatisch eingeblendet, und Sie sehen, dass im unteren Bereich die jeweiligen Feldeigenschaften erscheinen.


Allgemein	Nachschlagen
Feldgröße	255
Format	
Eingabeformat	
Beschriftung	
Standardwert	
Gültigkeitsregel	
Gültigkeitsmeldung	
Eingabe erforderlich	Nein
Leere Zeichenfolge	Ja
Indiziert	Nein
Unicode-Kompression	Ja
IME-Modus	Keine Kontrolle
IME-Satzmodus	Keine
Smarttags	

Abd. 3.19 Feldeigenschaften nach Eingabe des Feldnamens

- In die Spalte *Felddatentyp* wird nichts anderes eingetragen als der Felddatentyp. Die Bestimmung dieser verschieben wir auf Punkt 9.
- Als Beschreibung geben wir folgendes ein: *automatische Nummerierung der Studenten*. In Punkt 9 wird klar warum.
- Wenn Sie jetzt auf die TAB - TASTE oder die ENTER/RETURN-TASTE drücken, springt der Pfeil ganz vorne vor die Datenbeschreibung eine Zeile tiefer und Sie können die nächste Feldbeschreibung eingeben. Natürlich können Sie jederzeit in den alten Beschreibungen Änderungen vornehmen, indem Sie mit der Maus an die gewünschte Stelle klicken und dort - wie bei einer Textverarbeitung - löschen, einfügen und korrigieren.
- Füllen Sie die nächsten Feldbeschreibungen mit den folgenden Daten selber aus. In welcher Reihenfolge Sie das machen, ist beliebig:

Feldname	Felddatentyp	Beschreibung
ID_matrikel	Text	automatische Nummerierung der Studenten
txt_strasse	Text	Strasse des Studenten
txt_hausnummer	Text	Hausnummer des Studenten
txt_postleitzahl	Text	PLZ des Studenten
txt_wohnort	Text	Wohnort des Studenten
date_geburtsdatum	Text	Geburtstag des Studenten
byte_groesse	Text	Körpergröße des Studenten
txt_geschlecht	Text	Geschlecht des Studenten
txt_status	Text	Status: ordentliche eingeschriebener Student (oS), Zweithörer (ZH) oder Gasthörer (GH)
bool_studiengebuehren	Text	Muß der Student Studiengebühren bezahlen? Ja oder Nein

Abd. 3.20 Feldnamen der Tabelle: tbl\_studenten

- Wenn Sie diese Eintragungen soweit fertig haben, sollten Sie die Tabelle unbedingt einmal speichern. Am einfachsten geht das über das Diskettensymbol in der Symbolleiste für den Schnellzugriff oder über den Befehl **Speichern** aus dem Menübefehl **DATEI**. Oder Sie drücken die Tastenkombination STRG+S. Geben Sie als Dateinamen „tbl\_studenten“ ein.
- Als nächstes werden wir einen Primärschlüssel vergeben. Dafür kommt nur die Spalte Matrikelnummer in Frage (siehe Kapitel 1.1). Gehen Sie dazu in die Zeile Matrikelnummer und drücken Sie auf den Schlüssel  in der Multifunktionsleiste. Die Registerkarte **Entwurf** ist in der Entwurfsansicht automatisch ausgewählt. Danach sollte das Feld ‚Matrikelnummer‘ mit einem kleinen Schlüssel gekennzeichnet sein.

Feldname	Felddatentyp
ID_matrikel	Text
txt_strasse	Text

Abd. 3.21 Primärschlüssel mit Schlüsselssymbol gekennzeichnet

9. Nun wollen wir uns den Felddatentypen zuwenden. Welcher Datentyp soll den Matrikelnummern zugewiesen werden? Eigentlich eindeutig: *Access* soll automatisch weiterzählen, sobald ein neuer Datensatz eingegeben wird, womit gleichzeitig die Studenten durchnummeriert werden. Das kann nur der Datentyp „Auto-Wert“. Wir wählen also aus dem Dropdownfenster, in dem alle Felddatentypen aufgelistet sind, den Typ „Auto-Wert“ aus. „Auto-Wert“ hat die Eigenschaft, dass der Wert nur einmal vergeben werden kann. Exmatrikuliert sich ein Student bleibt seine Matrikelnummer gesperrt.

Feldname	Felddatentyp
ID_matrikel	AutoWert
txt_strasse	Text
txt_hausnummer	Memo
txt_postleitzahl	Zahl
txt_wohnort	Datum/Uhrzeit
date_geburtsdatum	Währung
byte_groesse	AutoWert
txt_geschlecht	Ja/Nein
txt_status	OLE-Objekt
bool_studiengebuehren	Hyperlink
	Nachschlage-Assistent

Abd. 3.22 Felddatentyp Auto-Wert

Welche Felddatentypen die weiteren Felder bekommen sollen und warum, können sie der folgenden Tabelle entnehmen. Die Begründungen sind in der folgenden Tabelle aufgeführt.

Feldname	Felddatentyp	Begründung
txt_strasse	Text	Straßennamen bestehen nun mal aus Text
txt_hausnummer	Text	Es ist zwar eine Zahl, aber man muss bestimmt nicht mit ihnen rechnen, und manchmal muss man Buchstaben eingeben, wie 2 B.
txt_postleitzahl	Text	Mit der PLZ muss man auch nicht rechnen.
txt_wohnort	Text	Schon mal einen Ort mit Zahlennamen gesehen, mit dem man womöglich noch rechnen muss? Nee.
dat_geburtsdatum	Datum/Uhrzeit	Logisch, oder?
txt_status	Text	Der Status besteht aus Buchstaben.
byte_groesse	Zahl	Klar, oder nicht ? Nähere Eigenschaften werden über die Feldeigenschaft bestimmt.
txt_geschlecht	Text	Es soll zwischen „männlich“ und „weiblich“ ausgewählt werden können.
bool_studiengebuehren	Ja/Nein	In das Feld soll ein Häkchen gesetzt werden können, was bedeutet dass der Student Gebühren zahlen muss.

Nun bestimmen wir die Eigenschaften der Felder genauer. Wie ? Über die *Feldeigenschaften*.

### 3.4 Bestimmung und Eingabe der Feldeigenschaften

#### Übung:

Überlegen Sie, welche Feldeigenschaften angegeben werden müssen, wenn

Im Feld *Auto-Wert* die Eingabe einer 7-stelligen Zahl zwingend sein soll, und die Stellen, die der Zähler nicht braucht mit einer 0 versehen werden. Außerdem sollen keine doppelten Werte, also zweimal dieselbe Matrikelnummer, vergeben werden dürfen. Das Ergebnis in der Tabelle könnte dann so aussehen: 0000001

Im Feld *Strasse* ausreichend Speicherplatz zur Verfügung steht.

Im Feld *Hausnummer* höchstens 4 Zeichen eingegeben werden dürfen.

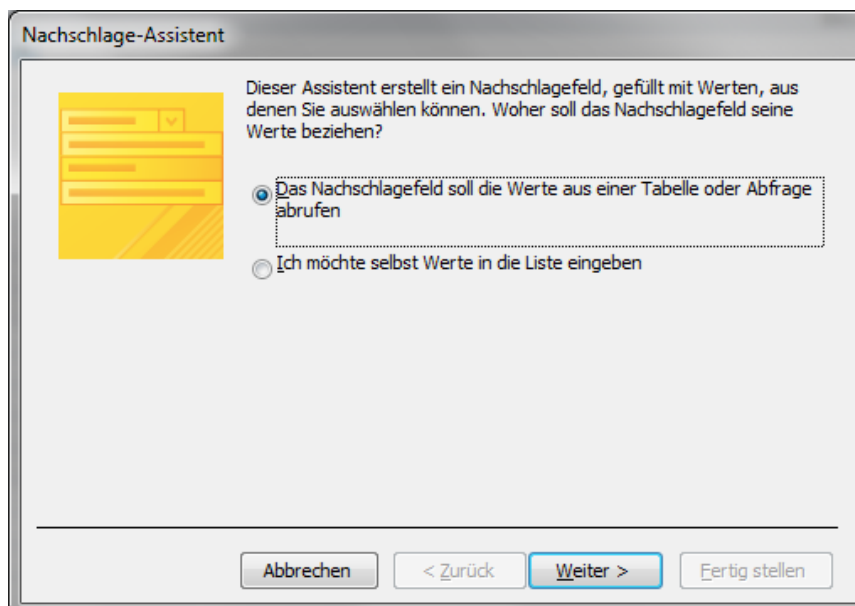
Im Feld *Postleitzahl* nur Zahlen, die aus nicht mehr und nicht weniger als 5 Ziffern bestehen, eingegeben werden können.

Im Feld *Wohnort* die eingegebenen Zeichen automatisch in Großbuchstaben umgewandelt werden.


Im Feld *Geburtsdatum* das passende Eingabeformat schon vorgegeben wird.

Im Feld *Körpergröße*, nicht mehr als 3 Zahlen eingegeben werden und automatisch das Maß „cm“ bei Eingabe des Wertes hinzugefügt wird.

Im Feld *Geschlecht* man zwischen „männlich“ und „weiblich“ aus einem Dropdownfenster auswählen kann. Hier kann man den Nachschlage-Assistenten benutzen, über den man solch ein Feld erstellen kann. Man aktiviert ihn, indem man ihn unter **Felddatentyp** aus dem Dropdownfenster auswählt. Folgendes Bild erscheint:



Abd. 3.23 Nachschlage-Assistent

Drückt man auf  bekommt man weitere Anweisungen zur Erstellung eines Dropdownfensters.

Im Feld *Status* nur die Zeichen oS für „ordentlicher Studierender“, ZH für „Zweithörer“ und GH für „Gasthörer“ erlaubt ist.

**Lösung**, die sie sich aber erst nach ihrer Übung anschauen sollten:

Feld	Feldeigenschaften																														
ID_matrikel	<table border="1"> <thead> <tr> <th>Allgemein</th> <th>Nachschlagen</th> </tr> </thead> <tbody> <tr> <td>Feldgröße</td> <td>Long Integer</td> </tr> <tr> <td>Neue Werte</td> <td>Inkrement</td> </tr> <tr> <td>Format</td> <td>0000000</td> </tr> <tr> <td>Beschriftung</td> <td>Matrikelnummer</td> </tr> <tr> <td>Indiziert</td> <td>Ja (Ohne Duplikate)</td> </tr> <tr> <td>Smarttags</td> <td></td> </tr> <tr> <td>Textausrichtung</td> <td>Standard</td> </tr> </tbody> </table>	Allgemein	Nachschlagen	Feldgröße	Long Integer	Neue Werte	Inkrement	Format	0000000	Beschriftung	Matrikelnummer	Indiziert	Ja (Ohne Duplikate)	Smarttags		Textausrichtung	Standard														
Allgemein	Nachschlagen																														
Feldgröße	Long Integer																														
Neue Werte	Inkrement																														
Format	0000000																														
Beschriftung	Matrikelnummer																														
Indiziert	Ja (Ohne Duplikate)																														
Smarttags																															
Textausrichtung	Standard																														
txt_strasse	<table border="1"> <thead> <tr> <th>Allgemein</th> <th>Nachschlagen</th> </tr> </thead> <tbody> <tr> <td>Feldgröße</td> <td>255</td> </tr> <tr> <td>Format</td> <td></td> </tr> <tr> <td>Eingabeformat</td> <td></td> </tr> <tr> <td>Beschriftung</td> <td>Straße</td> </tr> <tr> <td>Standardwert</td> <td></td> </tr> <tr> <td>Gültigkeitsregel</td> <td></td> </tr> <tr> <td>Gültigkeitsmeldung</td> <td></td> </tr> <tr> <td>Eingabe erforderlich</td> <td>Nein</td> </tr> <tr> <td>Leere Zeichenfolge</td> <td>Ja</td> </tr> <tr> <td>Indiziert</td> <td>Nein</td> </tr> <tr> <td>Unicode-Kompression</td> <td>Ja</td> </tr> <tr> <td>IME-Modus</td> <td>Keine Kontrolle</td> </tr> <tr> <td>IME-Satzmodus</td> <td>Keine</td> </tr> <tr> <td>Smarttags</td> <td></td> </tr> </tbody> </table>	Allgemein	Nachschlagen	Feldgröße	255	Format		Eingabeformat		Beschriftung	Straße	Standardwert		Gültigkeitsregel		Gültigkeitsmeldung		Eingabe erforderlich	Nein	Leere Zeichenfolge	Ja	Indiziert	Nein	Unicode-Kompression	Ja	IME-Modus	Keine Kontrolle	IME-Satzmodus	Keine	Smarttags	
Allgemein	Nachschlagen																														
Feldgröße	255																														
Format																															
Eingabeformat																															
Beschriftung	Straße																														
Standardwert																															
Gültigkeitsregel																															
Gültigkeitsmeldung																															
Eingabe erforderlich	Nein																														
Leere Zeichenfolge	Ja																														
Indiziert	Nein																														
Unicode-Kompression	Ja																														
IME-Modus	Keine Kontrolle																														
IME-Satzmodus	Keine																														
Smarttags																															
txt_hausnummer	<table border="1"> <thead> <tr> <th>Allgemein</th> <th>Nachschlagen</th> </tr> </thead> <tbody> <tr> <td>Feldgröße</td> <td>4</td> </tr> <tr> <td>Format</td> <td></td> </tr> <tr> <td>Eingabeformat</td> <td></td> </tr> <tr> <td>Beschriftung</td> <td>Hausnummer</td> </tr> <tr> <td>Standardwert</td> <td></td> </tr> <tr> <td>Gültigkeitsregel</td> <td></td> </tr> <tr> <td>Gültigkeitsmeldung</td> <td></td> </tr> <tr> <td>Eingabe erforderlich</td> <td>Nein</td> </tr> <tr> <td>Leere Zeichenfolge</td> <td>Ja</td> </tr> <tr> <td>Indiziert</td> <td>Nein</td> </tr> <tr> <td>Unicode-Kompression</td> <td>Ja</td> </tr> <tr> <td>IME-Modus</td> <td>Keine Kontrolle</td> </tr> <tr> <td>IME-Satzmodus</td> <td>Keine</td> </tr> <tr> <td>Smarttags</td> <td></td> </tr> </tbody> </table>	Allgemein	Nachschlagen	Feldgröße	4	Format		Eingabeformat		Beschriftung	Hausnummer	Standardwert		Gültigkeitsregel		Gültigkeitsmeldung		Eingabe erforderlich	Nein	Leere Zeichenfolge	Ja	Indiziert	Nein	Unicode-Kompression	Ja	IME-Modus	Keine Kontrolle	IME-Satzmodus	Keine	Smarttags	
Allgemein	Nachschlagen																														
Feldgröße	4																														
Format																															
Eingabeformat																															
Beschriftung	Hausnummer																														
Standardwert																															
Gültigkeitsregel																															
Gültigkeitsmeldung																															
Eingabe erforderlich	Nein																														
Leere Zeichenfolge	Ja																														
Indiziert	Nein																														
Unicode-Kompression	Ja																														
IME-Modus	Keine Kontrolle																														
IME-Satzmodus	Keine																														
Smarttags																															

Feld	Feldeigenschaften																														
txt_postleitzahl	<table border="1"> <thead> <tr> <th data-bbox="568 266 687 293">Allgemein</th> <th data-bbox="694 266 1166 293">Nachschlagen</th> </tr> </thead> <tbody> <tr> <td>Feldgröße</td> <td>5</td> </tr> <tr> <td>Format</td> <td></td> </tr> <tr> <td>Eingabeformat</td> <td></td> </tr> <tr> <td>Beschriftung</td> <td>PLZ</td> </tr> <tr> <td>Standardwert</td> <td></td> </tr> <tr> <td>Gültigkeitsregel</td> <td>Wie "####"</td> </tr> <tr> <td>Gültigkeitsmeldung</td> <td>Postleitzahlen bestehen immer aus 5 Ziffern</td> </tr> <tr> <td>Eingabe erforderlich</td> <td>Nein</td> </tr> <tr> <td>Leere Zeichenfolge</td> <td>Ja</td> </tr> <tr> <td>Indiziert</td> <td>Nein</td> </tr> <tr> <td>Unicode-Kompression</td> <td>Ja</td> </tr> <tr> <td>IME-Modus</td> <td>Keine Kontrolle</td> </tr> <tr> <td>IME-Satzmodus</td> <td>Keine</td> </tr> <tr> <td>Smarttags</td> <td></td> </tr> </tbody> </table>	Allgemein	Nachschlagen	Feldgröße	5	Format		Eingabeformat		Beschriftung	PLZ	Standardwert		Gültigkeitsregel	Wie "####"	Gültigkeitsmeldung	Postleitzahlen bestehen immer aus 5 Ziffern	Eingabe erforderlich	Nein	Leere Zeichenfolge	Ja	Indiziert	Nein	Unicode-Kompression	Ja	IME-Modus	Keine Kontrolle	IME-Satzmodus	Keine	Smarttags	
Allgemein	Nachschlagen																														
Feldgröße	5																														
Format																															
Eingabeformat																															
Beschriftung	PLZ																														
Standardwert																															
Gültigkeitsregel	Wie "####"																														
Gültigkeitsmeldung	Postleitzahlen bestehen immer aus 5 Ziffern																														
Eingabe erforderlich	Nein																														
Leere Zeichenfolge	Ja																														
Indiziert	Nein																														
Unicode-Kompression	Ja																														
IME-Modus	Keine Kontrolle																														
IME-Satzmodus	Keine																														
Smarttags																															
txt_wohnort	<table border="1"> <thead> <tr> <th data-bbox="568 651 687 678">Allgemein</th> <th data-bbox="694 651 1166 678">Nachschlagen</th> </tr> </thead> <tbody> <tr> <td>Feldgröße</td> <td>255</td> </tr> <tr> <td>Format</td> <td>&gt;</td> </tr> <tr> <td>Eingabeformat</td> <td></td> </tr> <tr> <td>Beschriftung</td> <td>Wohnort</td> </tr> <tr> <td>Standardwert</td> <td></td> </tr> <tr> <td>Gültigkeitsregel</td> <td></td> </tr> <tr> <td>Gültigkeitsmeldung</td> <td></td> </tr> <tr> <td>Eingabe erforderlich</td> <td>Nein</td> </tr> <tr> <td>Leere Zeichenfolge</td> <td>Ja</td> </tr> <tr> <td>Indiziert</td> <td>Nein</td> </tr> <tr> <td>Unicode-Kompression</td> <td>Ja</td> </tr> <tr> <td>IME-Modus</td> <td>Keine Kontrolle</td> </tr> <tr> <td>IME-Satzmodus</td> <td>Keine</td> </tr> <tr> <td>Smarttags</td> <td></td> </tr> </tbody> </table>	Allgemein	Nachschlagen	Feldgröße	255	Format	>	Eingabeformat		Beschriftung	Wohnort	Standardwert		Gültigkeitsregel		Gültigkeitsmeldung		Eingabe erforderlich	Nein	Leere Zeichenfolge	Ja	Indiziert	Nein	Unicode-Kompression	Ja	IME-Modus	Keine Kontrolle	IME-Satzmodus	Keine	Smarttags	
Allgemein	Nachschlagen																														
Feldgröße	255																														
Format	>																														
Eingabeformat																															
Beschriftung	Wohnort																														
Standardwert																															
Gültigkeitsregel																															
Gültigkeitsmeldung																															
Eingabe erforderlich	Nein																														
Leere Zeichenfolge	Ja																														
Indiziert	Nein																														
Unicode-Kompression	Ja																														
IME-Modus	Keine Kontrolle																														
IME-Satzmodus	Keine																														
Smarttags																															
date_geburtsdatum	<table border="1"> <thead> <tr> <th data-bbox="568 1126 687 1153">Allgemein</th> <th data-bbox="694 1126 1166 1153">Nachschlagen</th> </tr> </thead> <tbody> <tr> <td>Format</td> <td></td> </tr> <tr> <td>Eingabeformat</td> <td>00.00.0000;0;_</td> </tr> <tr> <td>Beschriftung</td> <td>Geburtsdatum</td> </tr> <tr> <td>Standardwert</td> <td></td> </tr> <tr> <td>Gültigkeitsregel</td> <td></td> </tr> <tr> <td>Gültigkeitsmeldung</td> <td></td> </tr> <tr> <td>Eingabe erforderlich</td> <td>Nein</td> </tr> <tr> <td>Indiziert</td> <td>Nein</td> </tr> <tr> <td>IME-Modus</td> <td>Keine Kontrolle</td> </tr> <tr> <td>IME-Satzmodus</td> <td>Keine</td> </tr> <tr> <td>Smarttags</td> <td></td> </tr> <tr> <td>Textausrichtung</td> <td>Standard</td> </tr> <tr> <td>Datumsauswahl anzeiger</td> <td>Für Datumsangaben</td> </tr> </tbody> </table>	Allgemein	Nachschlagen	Format		Eingabeformat	00.00.0000;0;_	Beschriftung	Geburtsdatum	Standardwert		Gültigkeitsregel		Gültigkeitsmeldung		Eingabe erforderlich	Nein	Indiziert	Nein	IME-Modus	Keine Kontrolle	IME-Satzmodus	Keine	Smarttags		Textausrichtung	Standard	Datumsauswahl anzeiger	Für Datumsangaben		
Allgemein	Nachschlagen																														
Format																															
Eingabeformat	00.00.0000;0;_																														
Beschriftung	Geburtsdatum																														
Standardwert																															
Gültigkeitsregel																															
Gültigkeitsmeldung																															
Eingabe erforderlich	Nein																														
Indiziert	Nein																														
IME-Modus	Keine Kontrolle																														
IME-Satzmodus	Keine																														
Smarttags																															
Textausrichtung	Standard																														
Datumsauswahl anzeiger	Für Datumsangaben																														




Feld	Feldeigenschaften																														
byte_groesse	<table border="1"> <thead> <tr> <th>Allgemein</th> <th>Nachschlagen</th> </tr> </thead> <tbody> <tr> <td>Feldgröße</td> <td>Byte</td> </tr> <tr> <td>Format</td> <td>000" cm"</td> </tr> <tr> <td>Dezimalstellenanzeige</td> <td>0</td> </tr> <tr> <td>Eingabeformat</td> <td></td> </tr> <tr> <td>Beschriftung</td> <td>Körpergröße</td> </tr> <tr> <td>Standardwert</td> <td>0</td> </tr> <tr> <td>Gültigkeitsregel</td> <td></td> </tr> <tr> <td>Gültigkeitsmeldung</td> <td></td> </tr> <tr> <td>Eingabe erforderlich</td> <td>Nein</td> </tr> <tr> <td>Indiziert</td> <td>Nein</td> </tr> <tr> <td>Smarttags</td> <td></td> </tr> <tr> <td>Textausrichtung</td> <td>Standard</td> </tr> </tbody> </table>	Allgemein	Nachschlagen	Feldgröße	Byte	Format	000" cm"	Dezimalstellenanzeige	0	Eingabeformat		Beschriftung	Körpergröße	Standardwert	0	Gültigkeitsregel		Gültigkeitsmeldung		Eingabe erforderlich	Nein	Indiziert	Nein	Smarttags		Textausrichtung	Standard				
Allgemein	Nachschlagen																														
Feldgröße	Byte																														
Format	000" cm"																														
Dezimalstellenanzeige	0																														
Eingabeformat																															
Beschriftung	Körpergröße																														
Standardwert	0																														
Gültigkeitsregel																															
Gültigkeitsmeldung																															
Eingabe erforderlich	Nein																														
Indiziert	Nein																														
Smarttags																															
Textausrichtung	Standard																														
txt_status	<table border="1"> <thead> <tr> <th>Allgemein</th> <th>Nachschlagen</th> </tr> </thead> <tbody> <tr> <td>Feldgröße</td> <td>2</td> </tr> <tr> <td>Format</td> <td></td> </tr> <tr> <td>Eingabeformat</td> <td></td> </tr> <tr> <td>Beschriftung</td> <td>Status</td> </tr> <tr> <td>Standardwert</td> <td></td> </tr> <tr> <td>Gültigkeitsregel</td> <td>In ("oS";"ZH";"GH")</td> </tr> <tr> <td>Gültigkeitsmeldung</td> <td>Hier dürfen nur folgende Zeichen verwendet werden: oS, ZH, GH</td> </tr> <tr> <td>Eingabe erforderlich</td> <td>Nein</td> </tr> <tr> <td>Leere Zeichenfolge</td> <td>Ja</td> </tr> <tr> <td>Indiziert</td> <td>Nein</td> </tr> <tr> <td>Unicode-Kompression</td> <td>Ja</td> </tr> <tr> <td>IME-Modus</td> <td>Keine Kontrolle</td> </tr> <tr> <td>IME-Satzmodus</td> <td>Keine</td> </tr> <tr> <td>Smarttags</td> <td></td> </tr> </tbody> </table>	Allgemein	Nachschlagen	Feldgröße	2	Format		Eingabeformat		Beschriftung	Status	Standardwert		Gültigkeitsregel	In ("oS";"ZH";"GH")	Gültigkeitsmeldung	Hier dürfen nur folgende Zeichen verwendet werden: oS, ZH, GH	Eingabe erforderlich	Nein	Leere Zeichenfolge	Ja	Indiziert	Nein	Unicode-Kompression	Ja	IME-Modus	Keine Kontrolle	IME-Satzmodus	Keine	Smarttags	
Allgemein	Nachschlagen																														
Feldgröße	2																														
Format																															
Eingabeformat																															
Beschriftung	Status																														
Standardwert																															
Gültigkeitsregel	In ("oS";"ZH";"GH")																														
Gültigkeitsmeldung	Hier dürfen nur folgende Zeichen verwendet werden: oS, ZH, GH																														
Eingabe erforderlich	Nein																														
Leere Zeichenfolge	Ja																														
Indiziert	Nein																														
Unicode-Kompression	Ja																														
IME-Modus	Keine Kontrolle																														
IME-Satzmodus	Keine																														
Smarttags																															
bool_studiengebuehren	<table border="1"> <thead> <tr> <th>Allgemein</th> <th>Nachschlagen</th> </tr> </thead> <tbody> <tr> <td>Format</td> <td>Ja/Nein</td> </tr> <tr> <td>Beschriftung</td> <td>Studiengebühren</td> </tr> <tr> <td>Standardwert</td> <td></td> </tr> <tr> <td>Gültigkeitsregel</td> <td></td> </tr> <tr> <td>Gültigkeitsmeldung</td> <td></td> </tr> <tr> <td>Indiziert</td> <td>Nein</td> </tr> <tr> <td>Textausrichtung</td> <td>Standard</td> </tr> </tbody> </table>	Allgemein	Nachschlagen	Format	Ja/Nein	Beschriftung	Studiengebühren	Standardwert		Gültigkeitsregel		Gültigkeitsmeldung		Indiziert	Nein	Textausrichtung	Standard														
Allgemein	Nachschlagen																														
Format	Ja/Nein																														
Beschriftung	Studiengebühren																														
Standardwert																															
Gültigkeitsregel																															
Gültigkeitsmeldung																															
Indiziert	Nein																														
Textausrichtung	Standard																														

Die übrigen Werte brauchen Sie nicht zu verändern, da dort **Access** die Standardeigenschaften verwenden darf. Jetzt speichern wir unsere fertige Datenstrukturtablelle entweder mit STRG + S oder mit dem Befehl **Speichern** aus der Symbolleiste für den Schnellzugriff.



### 3.4.1 Eingabe von Daten

Tragen Sie nun in die verschiedenen Felder entsprechende Datensätze ein bzw. auch mal einen Datensatz, der nicht den Feldeigenschaften entspricht. Sie werden merken, dass bei der einen und anderen Eingabe Fehlermeldungen erscheinen, die sie auf die falsche Dateneingabe hinweisen.

## Übung

- Als nächstes geben Sie bitte die folgenden Daten über die Studenten in Ihre Tabelle ein. Wie Sie an die Tabelle kommen? Ganz einfach! Klicken Sie oben links in der Multifunktionsleiste auf das Tabellensymbol . Zwischen der Entwurfs- und Datenblattansicht müssen Sie häufig wechseln, da es auch bei den anderen Teilen (Formular, Bericht) von **Access** diese Trennung in Entwurf und Ergebnis gibt.


Matrikelnummer	Straße	Hausnummer	PLZ	Wohnort	Geburtsdatum	Körpergröße	Geschlecht	Status	Studiengebühren
0000001	Kontzestr.	14	22233	DORTMUND	13.04.2004	123 cm	männlich	oS	<input checked="" type="checkbox"/>
0000002	Singenstr.	123	34565	DORTMUND	12.03.1900	145 cm	weiblich	ZH	<input checked="" type="checkbox"/>
0000003	Turmstr.	45	85904	MÜNCHEN	13.02.1749	178 cm	männlich	GH	<input type="checkbox"/>
0000004	Hippenstr.	90	29485	BREMEN	12.09.1790	145 cm	weiblich	GH	<input type="checkbox"/>
0000005	Teufelsstr.	333	98456	BERLIN	13.04.1979	167 cm	männlich	oS	<input checked="" type="checkbox"/>
0000006	Gebäudestr.	5	04895	BOCHUM	19.06.1980	187 cm	männlich	oS	<input type="checkbox"/>
0000007	Schnickstr.	67	83643	FREIBURG	12.03.1974	169 cm	weiblich	GH	<input checked="" type="checkbox"/>
*(Neu)						000 cm			<input type="checkbox"/>


- Dort wo Sie die Tabelle nicht genau lesen können, erfinden Sie sich einfach eigene Daten. Ansonsten können Sie bei der Eingabe alle Hilfsmittel, die Sie von der Textverarbeitung kennen, benutzen (Löschen, Einfügen, Umstellen etc.) Zum Schluss müssen Sie natürlich wieder speichern.
- Versuchen Sie bei der Eingabe einmal, gegen die Gültigkeitsregeln zu verstoßen, damit Sie sehen ob es klappt und das Programm auch die richtige Fehlermeldung ausgibt.
- Achten Sie am Rande auch einmal darauf, dass **Access** in der ersten Spalte der Tabelle seine Symbole wechselt. Sobald Sie anfangen zu schreiben, wird aus dem Sternchen  ein kleiner Bleistift , der nach dem Speichern des Datensatzes wieder verschwindet.

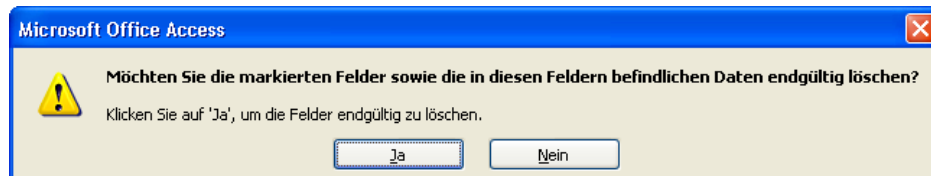
An dieser Stelle zwei **kleine Tricks**, die die Eingabe eventuell erleichtern: **Zum einen** kann man sich wiederholende Einträge aus der vorigen Zeile übernehmen, indem man STRG + Apostroph drückt. Aus der darüber liegenden Zeile wird dann der Zellinhalt einfach kopiert. **Zum anderen** können Sie sich jede Zelle, in die Sie irgend etwas eintragen möchten, vergrößern, indem Sie UMSCHALTEN + F2 drücken. Diese ZOOM-Funktion sollten Sie unbedingt einmal ausprobieren.

### 3.4.2 Ändern der Datenstruktur

Nicht immer kann man alle Eventualitäten der Datenstruktur schon auf einem Blatt Papier im Vorhinein abschätzen. Deshalb ist es wichtig, dass man auch später noch die Tabellen verändern kann. **Aber Vorsicht!** Bei dieser Art der Manipulation können immer auch Daten verloren gehen.

Um ein **neues Feld** der Tabelle hinzuzufügen, müssen Sie in der Entwurfsansicht die Zeile markieren, die sich unterhalb der *neuen* Zeile befindet. Nun wählen Sie den Befehl  **Zellen einfügen** aus der Multifunktionsleiste oder über einen rechten Mausklick. In dieser neuen Zeile können Sie dann ganz normal wieder ein Feld definieren.

Um ein Feld (und damit auch alle Daten, die in der Spalte stehen!!!) zu löschen, markieren Sie das Feld und wählen  **Zellen löschen** oder drücken einfach **ENTF**. Um wirklich sicher zu gehen, ob man den Datensatz *für immer* löschen will, fragt **Access** noch einmal nach. Die Meldung sieht dann folgendermaßen aus:



Abd. 3.24 Warnung vor dem endgültigen Löschen

## Übung

1. Erstellen Sie unterhalb der Zeile **byte\_groesse** ein neues Feld **txt\_augenfarbe** (oder irgendein anderes). Darunter noch eines für z.B. Haarfarbe. Die Feldgröße soll jeweils 30 betragen.
2. Schalten Sie zurück in die Tabellenansicht. **Access** merkt sofort, dass sich etwas geändert hat und fordert Sie zum Speichern auf. Das machen sie dann auch.
3. Jetzt können Sie hier eben Augen- und Haarfarbe der Studenten eintragen. Müssen sie aber nicht, da wir die Spalten gleich wieder löschen.
4. Löschen sie die beiden Zeilen wieder. Dazu müssen Sie sie in der Entwurfsansicht vorher markieren, was dann so aussieht, wie in Abd. 3.25 zu sehen.

Noch eins:

Auch um ein Feld innerhalb der Tabelle an eine **andere Spaltenposition** zu bugsieren, muss man die Zeile vorher markiert haben.



Abd. 3.25 Markierte Zeile

Danach packt man mit gedrückter linker Maustaste ganz vorne auf das Rechteck  und schiebt die ganze Zeile an die gewünschte Stelle.

## Übung






Versuchen Sie das Feld **Geburtsdatum** hinter **Körpergröße** zu platzieren, und wieder zurück.

### 3.4.3 Bewegen in den Tabellen (Navigationssymbole und Tastaturbelegung)

Damit Sie bei großen Datenblättern nicht mit dem Cursor oder Rollbalken als einzige Anspringmöglichkeit bestimmter Datensätze dastehen, gibt es ein paar Navigationssymbole am unteren linken Rand der Tabelle, die Ihnen dabei unter die Arme greifen.



Abd. 3.26 Navigationssymbole

Navigationssymbol	Wirkung
	Springt zum ersten Datensatz
	Springt zum letzten Datensatz
	Springt eine Zeile höher
	Springt eine Zeile tiefer
	Zeigt an, bei welchem Datensatz man sich befindet. Springt außerdem an den Datensatz, dessen Nummer man in das Kästchen schreibt

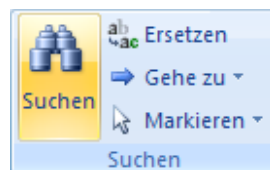
Außerdem gibt es natürlich auch wieder die Möglichkeit, mit der Tastatur die Sache in den Griff zu bekommen. Für alle Schnellschreiber ist das sicher die elegantere Art und Weise, sich in einer Tabelle fortzubewegen:

Tasten	Wirkung
TAB-Taste	Springt eine Zelle weiter
Umschalten + Tab	Springt eine Zelle zurück
Pos1	Springt auf die erste Zelle der aktuellen Zeile
Ende	Springt auf die letzte Zelle der aktuellen Zeile
Strg + Pos 1	Springt auf die erste Zelle der ersten Zeile
Strg + Ende	Springt auf die letzte Zelle der letzten Zeile
Strg + +	Aktiviert die erste Zelle des leeren Datensatzes, den <i>Access</i> automatisch anzeigt
Umschalten + F2	Öffnet das Zoom-Fenster

Probieren Sie es einfach einmal aus. Was? Funktioniert nicht? Ja, das kann sein. Man muss nämlich wissen, dass diese Tastenkombinationen nur funktionieren, wenn *Access* sich im **Überschreibmodus** befindet. Wenn es also nicht klappt, dann drücken Sie einfach **F2** und probieren die Übung noch einmal. Klappt's? Na sehen Sie.

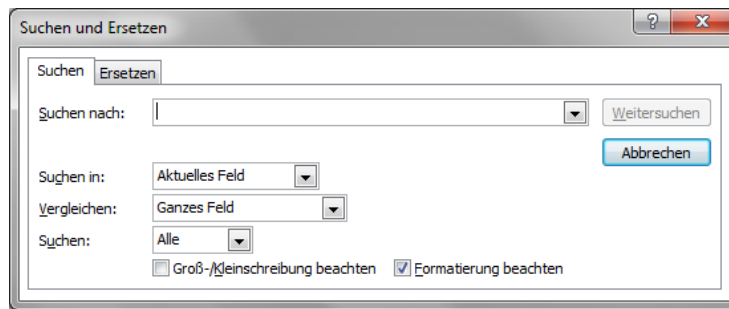
### 3.4.4 Datensätze in der Tabellenansicht suchen

Eigentlich ist die Tabellenansicht diejenige Aufbereitungsform der Daten, die der normale Anwender am häufigsten benutzt, obwohl das Handbuch von *Access* vorschlägt, sich mehr auf die Formulare (siehe nächstes Kapitel) zu stürzen. Eigentlich haben die Autoren recht, da man dort alles viel komfortabler einrichten kann. Aber manchmal ist der Mensch auch ein Gewohnheitstier. Deshalb kann man in der Tabellenansicht auch Datensätze suchen lassen. Entweder benutzt man das Suchfeld aus der Navigationsleiste  oder dieses nette Fernglas in der Multifunktionsleiste auf der Registerkarte Start:



Abd. 3.27 Symbol zum Suchen von bestimmten Datensätzen

Voraussetzung ist, dass man die Spalte, die den Suchbegriff enthält, markiert hat. Es reicht, wenn der Cursor dort irgendwo blinkt. Ansonsten kann man *Access* auch alle Felder durchsuchen lassen, aber das dauert entsprechend länger. Es öffnet sich ein Fenster, in das man sein gesammeltes Wissen eintragen kann:



Abd. 3.28 Suchen und Ersetzen-Fenster

Hier wurde der Cursor in die Spalte Matrikelnummer gesetzt. Diese Suchabfrage würde als Resultat den ersten Datensatz unserer Beispieltabelle ausgeben.

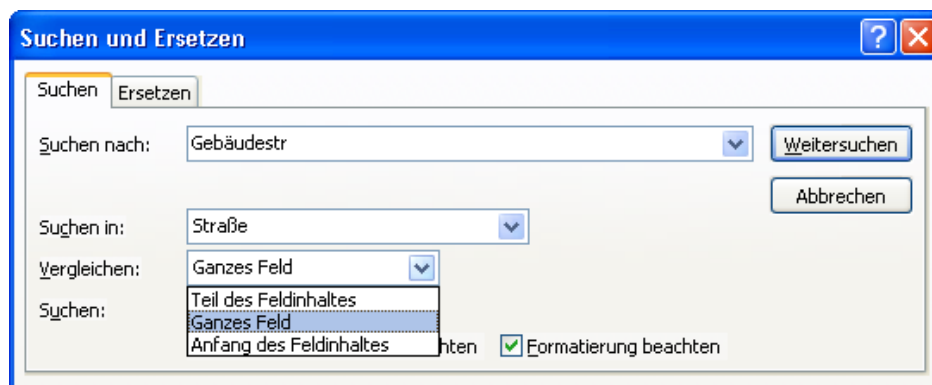
Um *Access* in der ganzen Tabelle suchen zu lassen, muss man in dem zu sehenden Dropdownfenster *tbl\_students* auswählen, also die komplette Tabelle und nicht nur eine Spalte. Wenn man weiß, in welcher Spalte der gesuchte Begriff vorkommt, setzt man den Cursor in die zu durchsuchende Spalte.

Drückt man auf *Weitersuchen* springt *Access* dann zu dem ersten gefundenen Begriff und unterlegt ihn. Wenn damit der Datensatz gefunden wurde, klickt man auf **Schließen**, ansonsten auf **Weitersuchen**.

## Übung

1. Probieren Sie das einmal aus, in dem Sie nach der Gebäudestr. suchen lassen.

Es gibt noch eine erweiterte Suchfunktion, die praktisch ist, wenn man nicht den kompletten Suchbegriff weiß, sondern nur einen Teil. Aus dem Dropdownfenster, welches sich in der Spalte *Vergleichen* öffnet, kann man *Teil des Feldinhaltes*, *Ganzes Feld* oder *Anfang des Feldinhaltes* auswählen, worüber man bestimmt welcher Teil eines Feldes nach dem eingegebenen Wort oder auch Wortteil durchsucht werden soll (s. **Abd.3.29**).



Abd.3.29 Suchen-Ersetzen-Fenster

Gibt man das Wortteil ‚berg‘ ein, findet *Access* bei

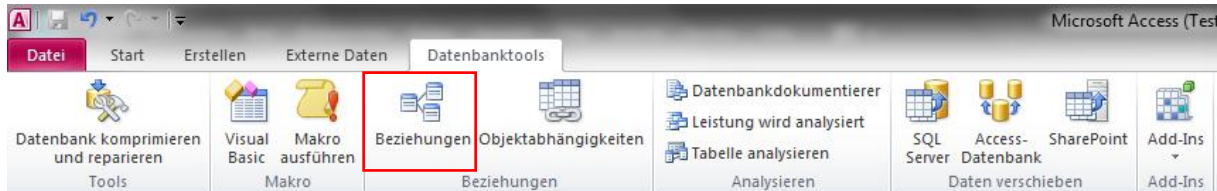
- *Teil des Feldinhaltes* sowohl Heidelberg, als auch Bergen
- *Anfang des Feldinhaltes* nur Bergen.

Gibt man das Wort ‚London‘ ein und wählt

- *Ganzes Feld*, findet *Access* nur ‚London‘ und nicht Londonderry‘

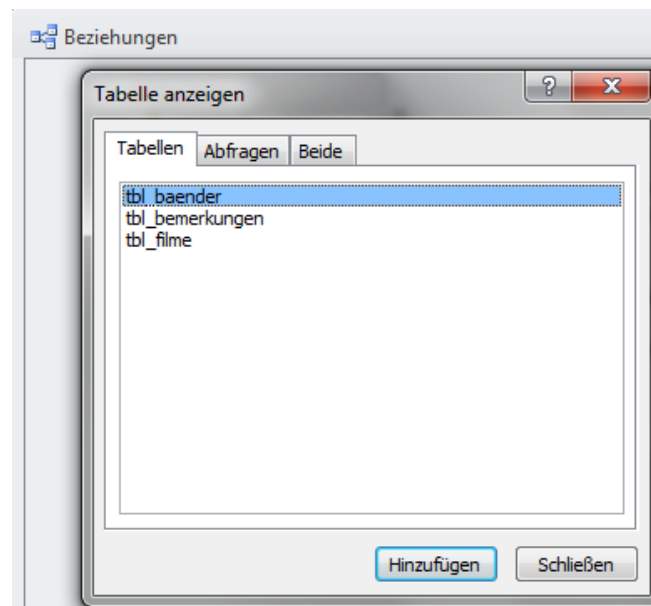
### 3.5 Tabellen verknüpfen

Wie schon mehrfach erwähnt wurde: Um Informationen aus mehreren Tabellen mischen zu können, muss man diese verknüpfen. Den Befehl, der dafür zuständig ist, findet man in der Multifunktionsleiste auf der Registerkarte **Datenbanktools** in der Befehlsgruppe **Beziehungen** und klickt dort auf *Beziehungen*.



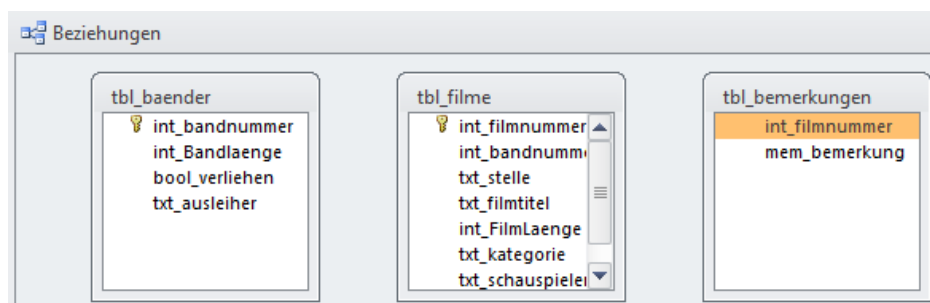
Abd. 3.30 Registerkarte Datenbanktools

Der sich öffnende Container erlaubt es, alle erstellten Tabellen dort einzufügen.



Abd. 3.31 Hier wählt man die Tabellen aus

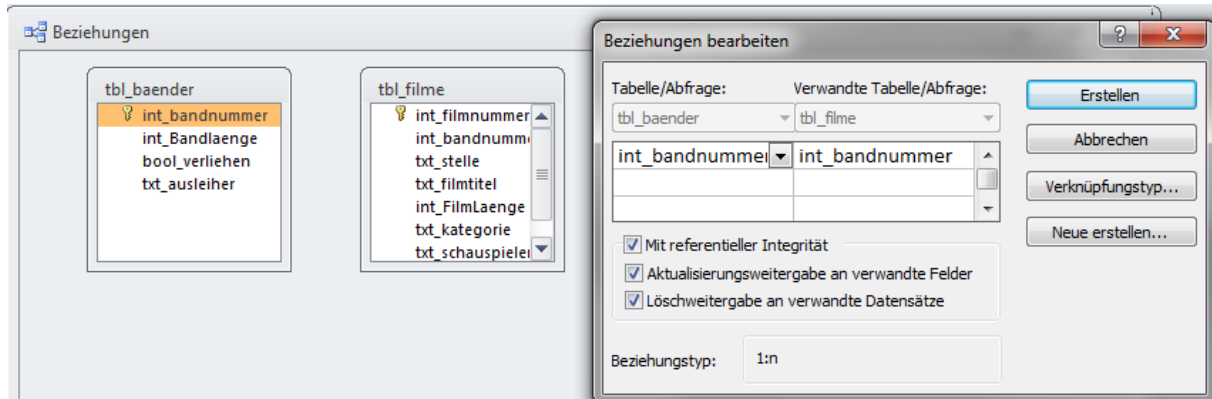
Im Endeffekt sieht das dann so aus:



Abd. 3.32 Und hier kann man sie dann wirklich verknüpfen

Der Rest ist einfach. Die mit einem Primärschlüssel versehenen Spalten erkennt man sofort an dem Schlüsselssymbol. Nun packt man einfach mit gedrückter linker Maustaste den Primärschlüssel der einen Tabelle und zieht ihn auf den Fremdschlüssel der zu verknüpfenden Tabelle. Damit diese Verknüpfung von **Access** akzeptiert wird, muss man darauf achten, dass die

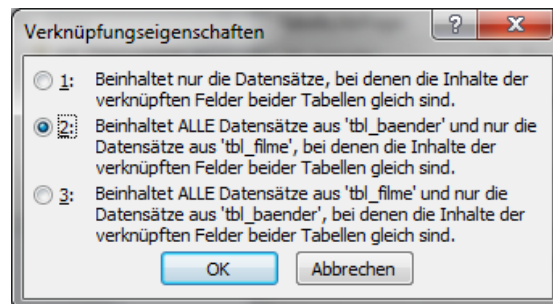
beiden Felder vom gleichen Typ sind, also *Text* oder *Zahl* oder *Datum*. Ist der Primärschlüssel in der *Mastertabelle* **Autowert** - zählt also automatisch hoch -, dann muss das verbundene Feld in der so genannten *Detailtabelle* vom Typ **Zahl /Long Integer** sein. Die Spaltennamen (int\_bandnummer) selber müssen allerdings nicht identisch sein, wie in unserem Fall. Nach dem Loslassen der linken Maustaste revanchiert sich *Access* mit folgender Nachfrage:



Abd. 3.33 Die referentielle Integrität sollte unbedingt angeklickt werden

Im oberen Teil zeigt das Programm die Spalten der Tabellen, die man verknüpfen möchte. Darunter sollte man unbedingt „Mit referentieller Integrität“ ankreuzen. Nur so überprüft *Access* die Konsistenz und Logik der Daten. Ob man nun möchte, dass Änderungen in der Mastertabelle ganz verboten werden oder an die Detailtabelle weitergereicht werden, hängt von der Logik der Anwendung ab. Bei einer Videothek ist die Löschweitergabe sicher berechtigt, da bei Verlust eines Bandes auch die Filme, die darauf sind, verschwunden sind. Geht es allerdings um das Ausscheiden eines Studenten aus der Uni, ist das nicht so günstig, da seine Daten ja eventuell später noch gebraucht werden.

Hinter dem Befehl *Verknüpfungstyp* verbirgt sich folgendes Fenster:



Abd.3.34 Fenster Verknüpfungseigenschaften

Was die einzelnen Eigenschaften (1,2,3) bewirken, ist dort erklärt, wobei *Felder mit gleichen Inhalten* nichts anderes meint, als dass sie Daten enthalten, also nicht leer sind.

Die häufigste Relation zwischen 2 Tabellen, die 1:n-Beziehung, versteckt sich hinter den Optionen 2 und 3, während Option 1 die 1:1-Beziehung beschreibt.

Wenn Sie bei der Erstellung einer 1:n-Beziehung stets zuerst den Primärschlüssel markieren und dann mit gedrückter linker Maustaste bis zum Fremdschlüssel ziehen, können Sie bei den Verknüpfungseigenschaften auch stets Option 2 wählen.



## 4 Abfragen

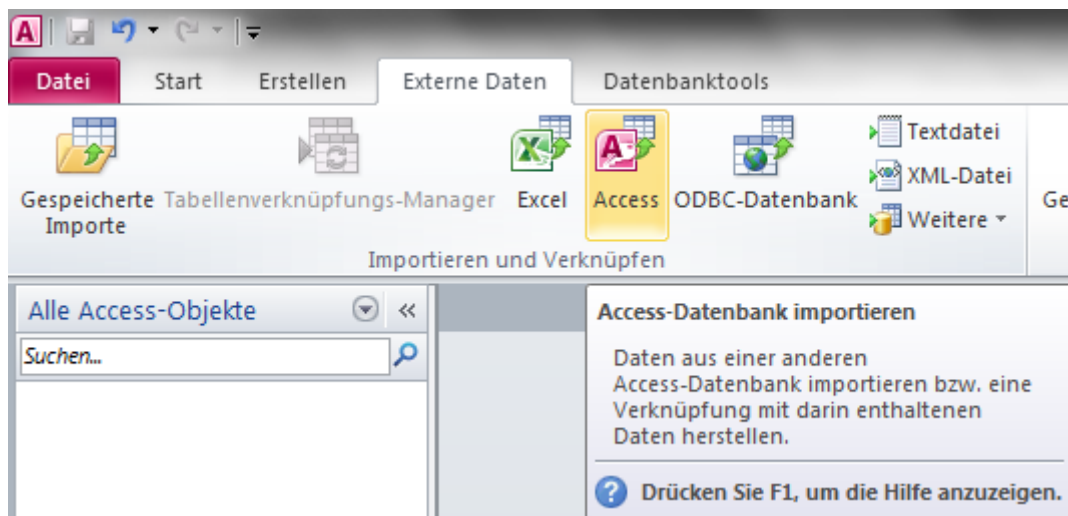
### 4.1 Trennen von Daten und Pflege derselben

Bevor man sich auf die weiteren Module des Datenbankmanagementprogramms stürzt, eine Vorbemerkung: Bevor man die erste Abfrage, den ersten Bericht oder das erste Formular bastelt, sollte man eine neue Datei erstellen. Es ist in *Access* günstig, die Datenbank mit den Tabellendaten zu trennen, von der Datenbankdatei, in der alles andere (Abfragen, Formulare, Berichte) gepflegt werden. So hat man jederzeit Zugriff auf das System, ohne die Dateneingabe zu stören. So schafft man es auch leicht, verschiedene Oberflächen und Abfragen für die verschiedenen Mitarbeiter zu erstellen, die aber alle wieder auf den gleichen Datenbestand zugreifen. Was ist dazu nötig?

Man erstellt eine neue leere Datenbank mit einem eindeutigen Namen. In diesem Skript wird in FrontEnd (FE) und BackEnd (BE) unterschieden. Im BackEnd befinden sich i.d.R. nur Tabellen, während sich Abfragen, Formulare und Berichte im FrontEnd befinden.

Für das Beispiel mit der Universitätsverwaltung wäre damit bspw. *univerwaltungFE.mdb* ein passender Dateiname. Dann weiß man später sofort, welche Datenbanken zueinander gehören.

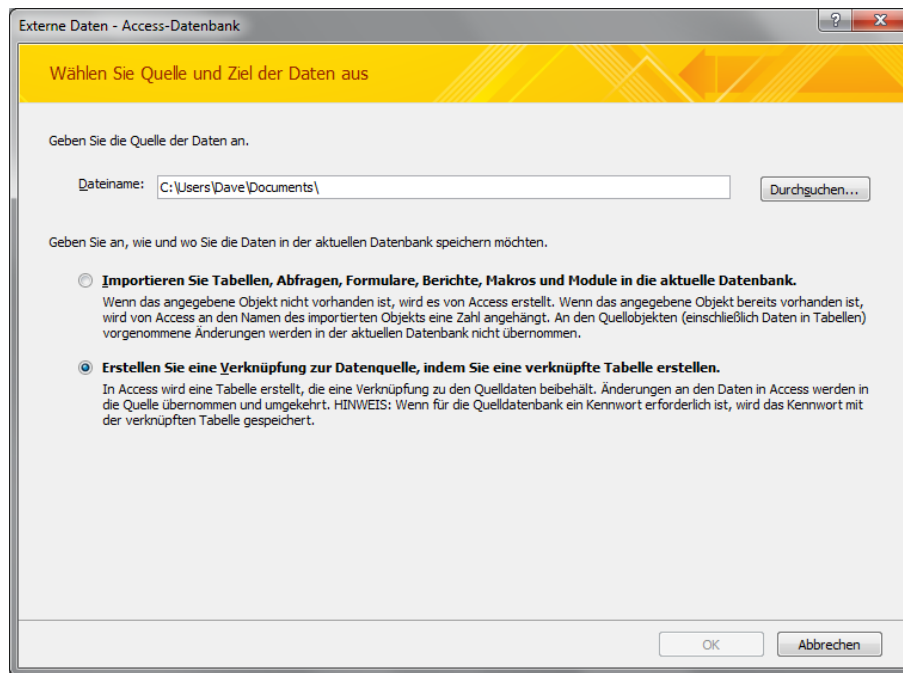
In den nächsten Beispielen bezieht sich das Prinzip auf eine Videofilmdatei. Entsprechend heißen die beiden Dateien *Videothek2010BE.accdb* und *Videothek2010FE.accdb*. Hat man die leere FE-Datei erstellt, wählt man in der Multifunktionsleiste auf der Registerkarte **Externe Daten** den Befehl *Access*:



Abd. 4.1 Verknüpfen von Tabellen

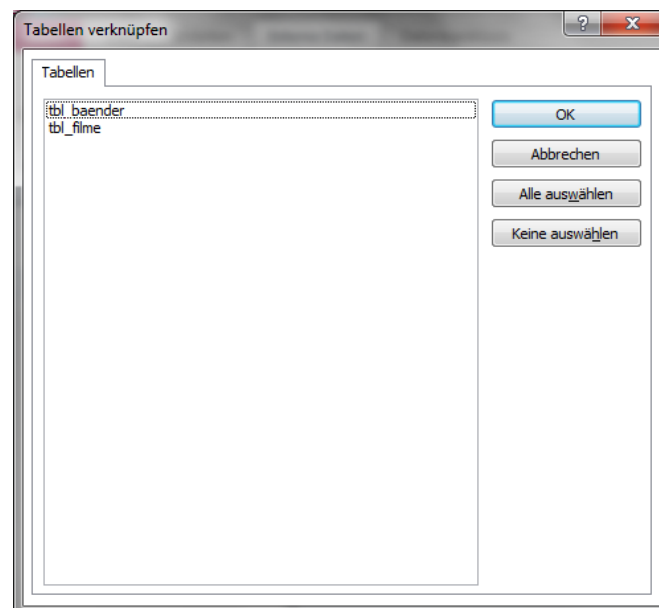
Darauf öffnet sich dann ein Dialogfenster, in dem man bestimmt aus welcher Datei die Daten stammen sollen sowie ob die Daten importiert oder verknüpft werden sollen.





Abd. 4.2 Aussuchen der Tab-Datei

Über die Schaltfläche *Durchsuchen...* kann die entsprechende Datei, in der sich die Tabellen befinden, ausgewählt werden. Und ein Doppelklick auf diese, zeigt die vorhandenen Tabellen an:



Abd. 4.3 Die Tabellen werden angezeigt

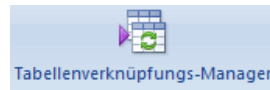
Mit dem Befehl *Alle Auswählen* und danach *OK* wird die Verknüpfung zur FE-Datei hergestellt. Man erkennt verknüpfte Tabellen an dem kleinen Pfeil, der vor dem Tabellennamen auftaucht.



Abd. 4.4 Die Pfeile weisen auf eine Verknüpfung hin

Alle Datensätze der Tabellen stehen einem nun zur Verfügung. Wenn man dort Daten einträgt, löscht oder ändert ist es so, als würde man dieses in der originalen Datenbank erledigen. Lediglich die Struktur der Daten kann man von hier nicht ändern, sondern dazu muss man tatsächlich die BE-Datei selber öffnen.

Verschiebt man die Tab-Datei, dann muss man in der FE-Datei *Access* erneut sagen, wo die verknüpften Tabellen liegen, und das auch, wenn beide Dateien im gleichen Verzeichnis bleiben. *Access* merkt sich nämlich leider den ganzen Pfad. Entweder löscht man schnell die Verknüpfung und fügt sie wieder neu ein, was ja kein großer Aufwand ist, oder man benutzt den Tabellenverknüpfungs-Manager. Der findet sich in der Multifunktionsleiste auf der Registerkarte **Datenbanktools** in der Befehlsgruppe *Datenbanktools*.



Abd. 4.5 Icon für den Tabellenverknüpfungs-Manager

Der erklärt sich von alleine. Und nun zurück zu den Abfragen:

## 4.2 Was leisten Abfragen?

Es ist bei der Einführung schon erwähnt worden, dass es Möglichkeiten der Datenaufbereitung gibt. Alles andere wäre auch langweilig. Das pure Sammeln von verschiedenen Informationen könnte man auch mit einer Textverarbeitung leisten. In *Access* heißt das dafür vorgesehene Werkzeug *Abfrage (query)*. Ihr Ergebnis, also die Antwort, nennt man *Dynaset*. Für Anwender anderer Datenbanken ist der Begriff *Sicht* bzw. *View* wahrscheinlich geläufiger. Nun, was geht dort alles?

- **Felder beschränken**, d. h. man trifft eine Auswahl der zu betrachtenden Spalten
- **Datensätze sortiert ausgeben lassen**, egal ob ab- oder aufsteigend
- **Datensätze beschränken**, d. h. man gibt Kriterien vor, welche Datensätze man sehen will, z. B. alle Krimis.
- **Daten aus mehreren Tabellen zusammenstellen**, d. h., dass man hier aus verschiedenen Tabellen Felder in einer Abfrage vereinen kann, was ja der Witz einer relationalen Datenbank ist.
- **Felder berechnen**, d. h. man kann in einer Abfrage neue Felder definieren, die durch eine Berechnung zustande kommen, z. B. *Bandlänge - Filmlänge = Restzeit*. Seit *Access 2010* gibt es auch in Tabellen die Möglichkeit, berechnete Felder einzufügen.
- **Datengrundlage für Formulare und Berichte und für andere Abfragen**, d. h. einmal erstellte Abfragen kann man behandeln wie normale Tabellen, für die man verschiedene Formulare und Berichte erstellen kann und die auch wieder als Grundlage für neue Abfragen dienen.

- **Neue Tabellen erstellen**, d. h., man kann die herausgefilterten Daten als eigenständige Tabelle speichern, die man dann in anderen Programmen benutzen kann. Wo braucht man so etwas? Richtig, bei Serienbriefen. Und das klappt wirklich recht flott.

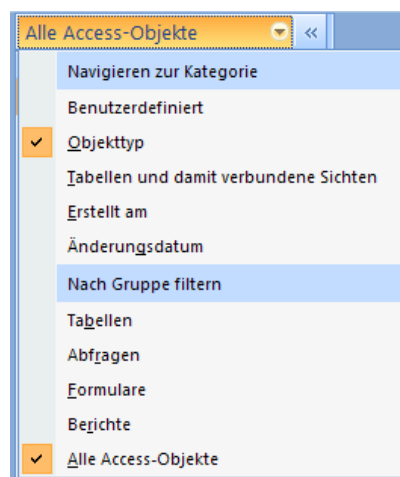
### 4.3 Wie erstellt man eine Abfrage?

Das Erstellen selber ist recht einfach, weil voll automatisiert. Beliebiger Schwierigkeit wird es erst, wenn die Selektions- und Verknüpfungsleistungen der Abfrage anspruchsvoller werden. D. h., das Grundwerkzeug ist schnell gelernt. Der Teufel steckt im Detail der konkreten Problemlösung. Aber was soll's!

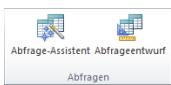
#### 4.3.1 Allgemeines


Es gibt mehrere Methoden eine Abfrage zu öffnen bzw. zu erstellen:

1. Bereits erstellte Abfragen lassen sich über den Navigationsbereich (früher: Datenbankfenster), welcher mit **F11** aus jeder *Access*-Lebenslage geöffnet werden kann, mit einem Doppelklick öffnen. Unter Umständen kann es nötig sein, die Ansicht im Navigationsbereich anzupassen um die bereits erstellten Abfragen zu sehen.

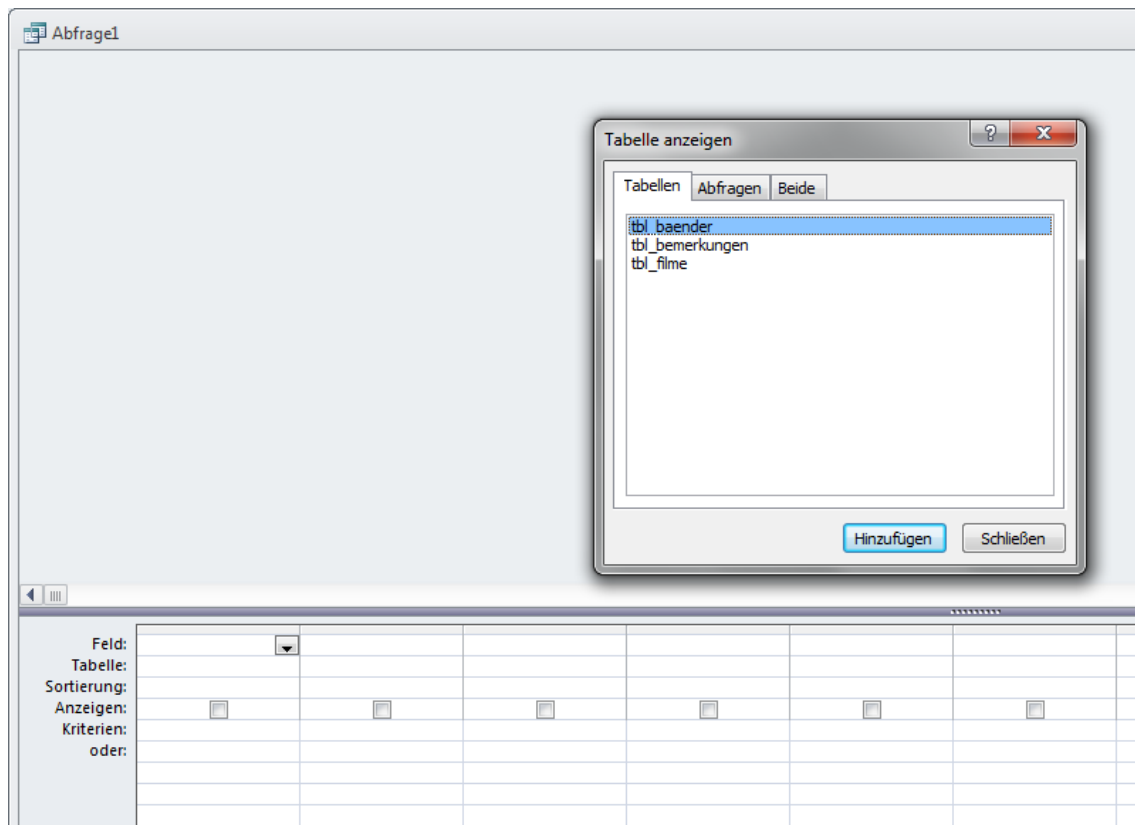


Abd. 4.6 Anzeige aller Objekte im Navigationsbereich

2. Drücken Sie auf das Symbol für den **Abfrage-Assistenten**  auf der Registerkarte **Erstellen** in der Multifunktionsleiste von *Access 2010* um eine neue Abfrage mit Hilfe des Assistenten zu erstellen.

3. Oder Sie klicken auf das Symbol **Abfrageentwurf**  um eine neue Abfrage in der Entwurfsansicht zu öffnen.

Da wir ja lernen wollen wie *Access* funktioniert, wählen wir hier den **Abfrageentwurf**. Darauf öffnet sich die *Entwurfsansicht* der Abfrage:



Abd. 4.7 Eine Leere Abfrage

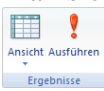
Im Fenster **Tabelle anzeigen** sind alle Tabellen aufgelistet, von denen man eine Abfrage erstellen kann. In Fall der **Videothek2010FE.accdb** sind es die **tbl\_baender**, die **tbl\_filme** und die **tbl\_bemerkungen**.

Der obere Teil der Entwurfsansicht ist *Auswahlbereich*, aus dem man Tabellenspalten auswählt. Der unter Bereich ist der *Entwurfsbereich*.

Die Logik sollte nun klar sein. Das Programm will eine oder mehrere Tabellen oder Abfragen als Grundlage für die neu zu erstellende Sicht. Mit den Knöpfen **Hinzufügen** und **Schließen** kann man ihm diesen Wunsch leicht erfüllen.

Danach kann man dann die gewünschten Felder auswählen und Kriterien bestimmen, nach denen diese selektiert werden sollen.

In einem letzten Schritt kann man sich die Ergebnisse - neudeutsch *Dynaset* - anschauen, in-

dem man über diese Buttons  in die Tabellenansicht wechselt oder die Abfrage ausführt. Beides führt zu demselben Resultat.

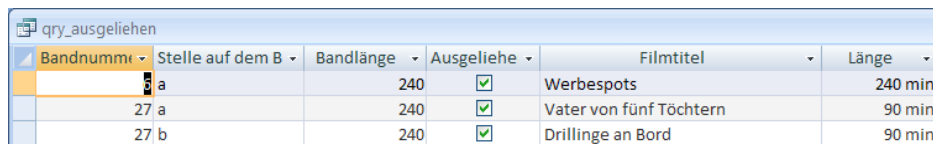
Will man an den Kriterien der Abfrage etwas ändern, muss man wieder zurück in die Entwurfsansicht. Das Prinzip der Abfrage besteht darin, dass das Dynaset jedes mal bei Aufruf der Tabellenansicht neu erstellt wird. Deshalb kann man auch beliebig viele Abfragen basteln, ohne dass man Angst haben muß, dass die Festplatte irgendwann in die Knie geht. Speicherplatz wird nur für die Abfrage, nicht aber für die Daten gebraucht.

Außerdem kann man die Abfrage auch noch speichern, z. B. mit STRG + S oder mit dem Symbol **Speichern** aus der Symbolleiste für den Schnellzugriff. Nach Vergabe eines sinnvollen Namens, erscheint sie im Navigationsbereich unter den Abfragen und ist jederzeit wieder verwendbar, z. B. auch, um dafür bzw. daraus ein Formular zu basteln (Kapitel 5.0)

### 4.3.2 Eine Abfrage und ihr Dynaset - Eine Frage und ihre Antwort

Eine Abfrage ist eine Frage, die der Benutzer über die Daten in seiner Datenbank stellt, wie z. B. "Welcher Regisseur hat den Film "Die Vögel" gedreht?". Die Daten, die Antwort auf diese Frage geben, stammen aus mehreren Tabellen. Eine Abfrage bringt die angeforderten Informationen zusammen.

Das Abfrageergebnis (Abd. 4.8) - die Gruppe von Datensätze, aus denen es besteht - wird Dynaset genannt. Im Beispiel werden die ausgeliehenen Bänder in der Videothek angezeigt.



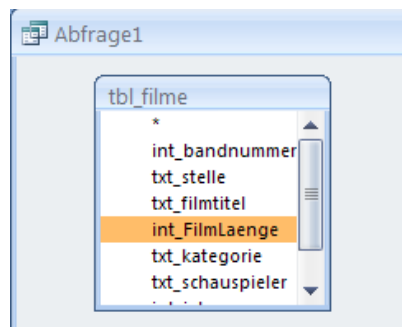
Bandnummer	Stelle auf dem B	Bandlänge	Ausgeliehe	Filmtitel	Länge
a	27	240	<input checked="" type="checkbox"/>	Werbespots	240 min
a	27	240	<input checked="" type="checkbox"/>	Vater von fünf Töchtern	90 min
b	27	240	<input checked="" type="checkbox"/>	Drillinge an Bord	90 min

Abd. 4.8 Dynaset (Abfrageergebnis)

### 4.3.3 Spezielles: Beispiele, Übungen und Tricks

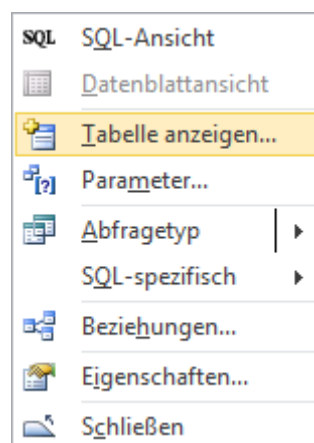
#### 4.3.3.1 Felder hinzufügen und entfernen

Es wurde schon gesagt, dass man nicht alle Spalten einer Tabelle ins Dynaset übernehmen muss, sondern selektieren kann. Voraussetzung dafür ist, dass im Auswahlbereich der Entwurfsansicht die Liste der Feldnamen einer oder mehrerer Tabellen zu sehen sind (s. Abd. 4.9).



Abd. 4.9 Feldnamenliste im Auswahlbereich

Falls das nicht der Fall ist, kann man nach einem Rechtsklick im oberen Bereich des Abfrageentwurfs den Befehl **Tabelle anzeigen...** auswählen und den Mangel beheben.


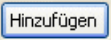


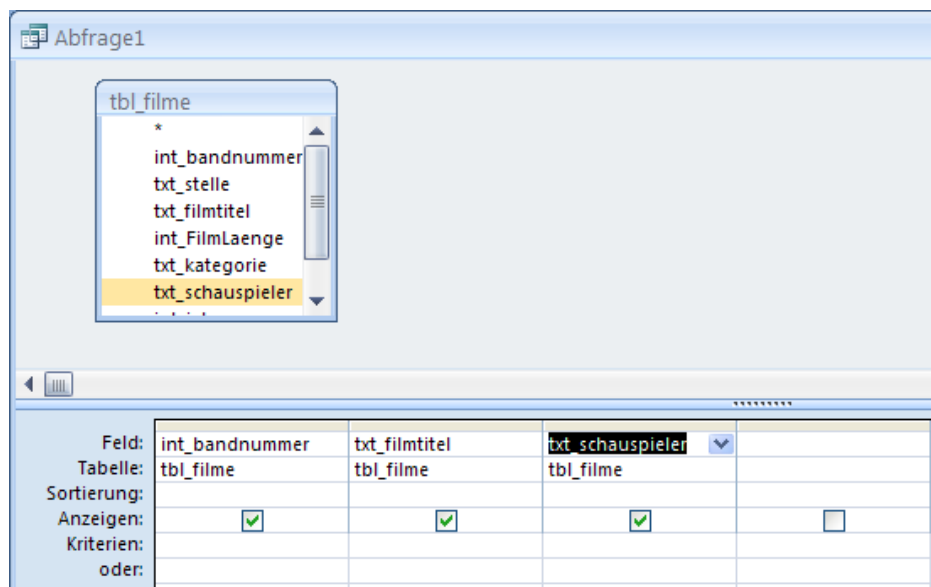
Abd. 4.10 Menü nach rechtem Mausklick im Abfrageentwurf

Danach kann man mit Doppelklick auf die gewünschten Feldnamen diese in die Entwurfsansicht einfügen. Sie erscheinen dann unten in der Entwurfsansicht der Abfrage.





Um das Prinzip etwas zu vertiefen und ein paar Tricks unterzubringen, vollziehen Sie bitte folgende Übung nach.

## Übung

1. Erstellen sie eine neue Abfrage in der Entwurfsansicht.
2. Die Nachfrage, mit welcher Tabelle Sie arbeiten wollen, beantworten Sie, indem Sie auf Filme und danach auf  klicken. So ein Mist, das Programm hat es nicht verstanden. Die obere Hälfte bleibt leer! Was tun? Wir gehen einfach über den Befehl *Tabelle anzeigen...* und wählen noch einmal aus. Was haben wir gelernt? Ohne den Knopf  weigert sich *Access*, die Tabelle mit aufzunehmen. Noch einfacher ist übrigens ein Doppelklick auf den Tabellennamen, aber das nur am Rande. Warum einfach, wenn es im Notfall auch kompliziert geht?
3. Ziehen Sie mit der Maus oder durch Doppelklick die Feldnamen `int_bandnummer`, `txt_filmtitel` und `txt_schauspieler` nach unten in die Entwurfsansicht.



Abd. 4.11 Die erste Abfrage

4. Starten Sie die Abfrage indem Sie auf  in der Symbolleiste klicken. Sie können auch den Button für die Tabellenansicht benutzen, der ganz links sitzt. Als Ergebnis erhalten Sie Ihr erstes Dynaset. Bravo!
5. Wechseln Sie zurück in die Entwurfsansicht durch Klicken auf  und fügen Sie die Filmlänge hinzu. Wieder mit  überprüfen.
6. Löschen Sie `txt_schauspieler`, indem Sie die Spalte markieren (oben auf die Kopfleiste drücken) und die ENTF-Taste drücken. Wieder mit überprüfen.
7. Löschen Sie die restlichen Feldnamen wie in Punkt 6. Doppelklicken Sie im oberen Teil des Fensters auf die Kopfleiste Ihrer **tbl\_filme**. Alle Feldnamen sollten danach markiert sein. Ziehen Sie mit der Maus den ganzen Sums nach unten. Ergebnis? **Alle Feldnamen** sind in die Abfrage übernommen worden.

8. Klicken Sie auf die Kopfzeile der Spalte int\_jahr (mit Kopfzeile meine ich diesen Teil



int_jahr	tbl_filme

der Spalte), so dass diese komplett markiert ist. Ziehen Sie die Spalte nach links. Mit dieser Methode können Sie Feldnamen einfach und schnell neu anordnen.

#### 4.3.3.2 Datensätze sortieren

Um Datensätze zu sortieren, müssen Sie erst einmal bestimmen, welche Spalten Sie in die Abfrage übernehmen wollen. Nur über diese können Sie auch sortieren, logo! Wir werden unser fertiges Beispiel von der vorigen Übung nehmen.

**EXKURS ZUM INDEX** Aber vorher noch ein Hinweis. Für das Ordnen der Datensätze sollten Sie für das Feld, **das das Kriterium abgibt**, einen Index setzen. Wir haben beim Herstellen der Tabellen bereits einen Primärschlüssel vergeben müssen, weil **Access** uns beim Speichern dazu gezwungen hat. Dieser Primärschlüssel war nichts anderes als ein Index mit besonderen Eigenschaften. Aber was soll das Ganze? Weil es so schön dort erklärt ist, möchte ich ausnahmsweise einmal in einem Skript zitieren (man muss ja nicht alles neu erfinden!):

*"Die Datensätze einer Tabelle werden hintereinander in eine Datei geschrieben. Bei großen Tabellen mit sehr vielen Datensätzen können so zwei Probleme entstehen:*

- *Wenn die Tabelle nach einem bestimmten Kriterium sortiert werden soll, müssen sehr viele Daten bewegt werden, bis die Tabelle in der neuen Reihenfolge vorliegt, weil jeder Datensatz mit allen Feldern umsortiert werden muss.*
- *Bei der Suche nach einem bestimmten Datensatz muss die Tabelle von vorne nach hinten durchsucht werden. Der Zeitaufwand ist bei großen Tabellen erheblich, weil bei der Suche vom Anfang bis zum Ende der Tabelle jeder Datensatz überprüft werden muss.*

*Beide Probleme kann man durch Indizieren eines bestimmten Feldes, nach dem sortiert oder gesucht werden soll, umgehen. Oft werden dazu von Programmen so genannte Indexdateien angelegt. In einer Indexdatei ist neben dem Feld, nach dem sortiert oder gesucht werden soll, nur ein Verweis auf den Datensatz eingetragen. Beim Sortieren von Daten muss somit viel weniger Datenmaterial bewegt werden und für das Suchen nach bestimmten Datensätzen gibt es in den Indexdateien spezielle Suchalgorithmen. Bei **Access** befinden sich die Tabelle und der Index zwar in der gleichen Datei, aber die Funktionsweise ist identisch.*

*Das Indizieren von Feldern hat allerdings auch seinen Preis. Da bei der Eingabe von Datensätzen für das indizierte Feld jeweils der Index angelegt werden muss, wird dadurch die Dateneingabe entsprechend verlangsamt.*

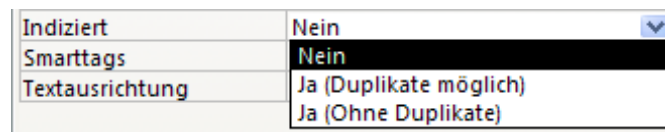
*Man unterscheidet zwei Arten der Indizierung:*

- **Primärschlüssel**  
*Ein Wert des Feldes, für den Primärschlüssel vergeben wurde, ist nur genau einmal vorhanden, wie beispielsweise die Filmnummer, denn jeder Film bekommt eine eindeutige Filmnummer.*
- **Sekundärschlüssel**  
*Ein Wert eines Feldes, für das ein Sekundärschlüssel vergeben wurde, kann mehrfach vorhanden sein. In einer Adressdatei beispielsweise kann es mehrere "Müller" geben. Eine eindeutige Zuordnung ist nicht möglich, aber ein schnelles Suchen nach einem bestimmten Namen ist durch die Indizierung trotzdem gegeben."<sup>1</sup>*

<sup>1</sup>Ralf Albrecht, Natascha Nicol: **Das Microsoft Access 1.1 Einmaleins**, Düsseldorf 1993, Seite 64 ff




Einen Index erstellt man in der gleichen Maske, in der man die Datenstruktur bestimmt, also in der Entwurfsansicht einer Tabelle. Dort war die letzte angesprochene Feldeigenschaft **Indiziert**. Öffnet man die Liste erscheint (vgl. 3. Kapitel):

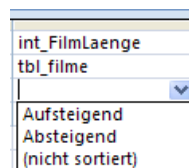


Abd. 4.12 Indexfenster



Durch entsprechendes Anklicken vergibt man so einen Primärschlüssel, einen sekundären Index oder löscht die entsprechende Einstellung.


## Übung

1. Vergeben Sie in der Tabellenentwurfsansicht einen *sekundären Index* für Filmlänge. Wechseln Sie danach wieder in unsere allgemeine Abfrage. Sie können dazu beide Fenster gleichzeitig geöffnet lassen. Ansonsten müssen Sie die Abfrage speichern. **Access** fragt automatisch danach, wenn Sie das Fenster schließen wollen. Nennen Sie es **qry\_erste\_aufgabe**.
2. Sortieren Sie Filme nach der Filmlänge und zwar absteigend. Dazu klicken Sie in die Zeile Sortierung in der Spalte int\_FilmLaenge. Es erscheint das altbekannte Dropdownfenster, wenn sie auf  klicken:



Abd. 4.13 Sortiermöglichkeiten

3. Suchen Sie sich "Absteigend" aus und lassen sich das Dynaset anzeigen. Wenn alles richtig gelaufen ist, dann steht der längste Film ganz oben, der kürzeste ganz unten.
4. Um über einem bestimmten Feld zu sortieren, muss das Feld zwar unten in der Entwurfsansicht stehen, aber es muss nicht unbedingt auch im Dynaset erscheinen. Um das ein- bzw. auszuschalten müssen Sie das Kästchen unter der Zeile Sortierung  Anzeigen:   , vor dem Anzeigen steht, anklicken. Ein Häkchen lässt die Spalte in der Sicht erscheinen, ohne Häkchen kann man es zwar für alle Funktionen benutzen aber die Daten werden nicht in Abfragetabelle gezeigt. Probieren Sie das für die Filmlänge mal aus.
5. Sortieren Sie nach einem weiteren Kästchen. D. h. Sie möchten die Filme nicht nur nach ihrer Länge sortiert haben, sondern zuerst nach ihrem Entstehungsjahr und anschließend erst nach der Länge. Wie? **Access** nimmt in einem solchen Fall die erste Spalte mit Sortierungskriterium als *primäre Sortierung*, die Spalte danach als *sekundäre* usw. In unserem Fall ist das primäre Kriterium die *Filmlänge* und das sekundäre das *Jahr*. Sie müssen also die Spalte **Jahr** links neben die Spalte **Filmlänge** verschieben (siehe oben) und für beide Spalte eine Sortierung auswählen.

Drückt man  **Ausführen**, kann man zu folgendem Ergebnis kommen:



Bandnumm.	Filmtitel	Jahr	Länge
57	Die Wahrheit über die Bounty	1994	45 min
56	Frankreichs Atombomben	1993	85 min
52	Gottschalks Oldie Night 93	1993	105 min
67	Nackte Kanone 2 1/2	1993	105 min
69	Robin Hood	1993	180 min
73	Der mit dem Wolf tanzt	1993	255 min
66	Kleine Haie	1992	90 min
52	Gottschalks Oldie Night 92	1992	105 min
74	Alien 3	1992	125 min
83	Boomerang	1992	135 min
70	Basic Instinct	1992	145 min
70	Allein unter Frauen	1991	90 min
83	Oscar - Vom Regen in die Traufe	1991	105 min
12	Das Geld anderer Leute	1991	125 min

Abd. 4.14 mögliches Ergebnis der Sortierung

Das soll zu diesem Thema reichen. Bei den nächsten Beispielen werden wir allerdings diese Möglichkeiten immer wiederholen, ohne dass extra der Weg neu beschrieben wird. So übt sich der Dreh mit der Zeit schon ein.

#### 4.3.3.3 Datensätze beschränken

Diese Funktion ist sicherlich eine der wichtigsten Vorteile einer Datenbank. Gezielt kann man sich nur die Datensätze anschauen, die bestimmten Kriterien genügen. Diese Kriterien sind sehr ausgefeilt definierbar. "Alle braunhaarigen über 1,90 m, die nicht in Köln oder Hamburg leben, ledig sind, jünger als 25, nicht mehr als 1 Kind und bis 1992 zwischen 1 500 € und 2 500 € netto verdient haben." oder so ähnlich.

Die Logik ist immer gleich: Sie schreiben in das für Sie interessante Feld die Bedingung in den Abfrageentwurf. Falls eine andere Bedingung in einem anderen Feld ebenfalls gelten soll, dann schreiben Sie sie in der **gleichen Zeile** in das entsprechende Feld, wenn Sie das logische UND brauchen, **eine Zeile tiefer** wenn Sie eine Verknüpfung mit dem logischen ODER benötigen. Zu abstrakt? Hier ein paar Beispiele:

- Der Film soll länger sein als 90 Minuten UND ein Krimi -> logisches UND
- Der Film soll entweder ein Krimi ODER ein Drama sein -> logisches ODER
- Der Film soll NICHT ein Krimi sein-> logisches NICHT

Natürlich kommt es hier wieder sehr auf die korrekte Schreibweise dieser **Ausdrücke** - so heißt das in **Access** - an, damit das Programm weiß, was man überhaupt von ihm will. Die folgenden Tabellen geben Aufschluss über die richtige Notation und die Möglichkeiten, die **Access** hier bietet:

#### Komponenten eines Ausdrucks

Komponente	Beispiel	Beschreibung
Operator	+, =, >, Oder, Wie	Führt an einer oder mehreren anderen Komponenten eine Operation aus.
Bezeichner	[Postleitzahl] Formulare!Aufträge!AuftragsNr	Gibt ein Feld, ein Steuerelement, ein Formular, eine Eigenschaft etc. an.
Funktion	DATUM, SUMME, DOMANZAHN	Gibt einen Funktionswert aus. Zulässig sind auch in <b>Access</b> -Basis geschriebene benutzerdefinierte Funktionen

Komponente	Beispiel	Beschreibung
Literal	100, "Hamburg", #1.1.93#	Zahl, Datumsangabe oder Text. Ein Literal wird ohne Interpretation zeichengetreu übernommen.
Konstante	Wahr, Falsch, Ja, Nein, Null	Fester Wert, der eine Zahl oder ein Text sein kann.

### Stellvertreterzeichen

Zeichen	Beispiel	Bedeutung
*	ma* findet man, Mann und Mauer *er findet Heiner, einer und Schornsteinfegermeister	Das Sternchen wirkt wie das DOS-Stellvertreterzeichen *, d. h. es kann einer beliebigen Anzahl Zeichen entsprechen. Im unterschied zu DOS kann es in <i>Access</i> jedoch auch für das erste oder letzte Zeichen stehen.
?	R?gen findet Regen und Rügen	Steht für ein beliebige einzelnes Zeichen; wirkt wie bei DOS.
#	l#3 findet 103,113,123	Steht für eine beliebige einzelne Zahl
[]	R[eü]gen findet Regen und Rügen , nicht jedoch Rogen	Steht für eines der innerhalb der eckigen Klammern angegebenen Zeichen.
!	R[!eü]gen findet Rogen, nicht jedoch Regen und Rügen	Steht für eines der innerhalb der eckigen Klammern nicht angegebenen Zeichen.
-	b[a-c]d findet bad, bbd und bcd	Steht für ein einzelnes, beliebiges Zeichen innerhalb des angegebenen Bereichs.

### Vergleichsoperatoren

Operator	Bedeutung
<	Kleiner als
<=	Kleiner als oder gleich
>	Größer als
>=	Größer als oder gleich
=	Gleich
<>	Ungleich

### Verkettungsoperatoren

Operator	Bedeutung
&	Verkettet zwei Operanden

## Logische Operatoren

Operator	Bedeutung
Und	Logisches Und
Nicht	Logisches Nicht
Oder	Inklusives Oder
ExOder	Exklusives Oder

## Weitere Operatoren

Operator	Bedeutung
Ist Null	Prüft, ob ein Wert Null ist oder nicht Null ist
Wie	Prüft, ob ein Zeichenfolgenausdruck einem als Suchbegriff angegebenen Zeichenfolgenausdruck gleicht; dabei dürfen Stellvertreterzeichen angegeben werden.
Zwischen ... Und In	Prüft, ob ein Wert zwischen zwei Grenzen liegt. Prüft, ob ein Wert in einer Wertliste vorkommt.

## Arithmetische Operatoren

Operator	Bedeutung
*	Multiplikation
+	Addition
-	Subtraktion oder Vorzeichenumkehr
/	Division von Gleitkommazahlen
\	Division von Integerzahlen
^	Potenzieren
Mod	Division zweier Zahlen und Ausgabe des Rests

Das ist viel Zeugs und noch nicht einmal alles. In dem Begleithandbuch von MicroSoft *Access* findet man noch ein paar mehr. Wir wollen im Weiteren die Sache mit ein paar Beispielen in den Griff bekommen, die Sie bitte gleich ausprobieren.

## Übung

Voraussetzung für die nun folgenden Übungen ist, dass Sie sich in der Entwurfsansicht der Abfrage befinden und alle Felder der Tabelle Filme aus dem Auswahlbereich in den Entwurfsbereich geklickt haben.

1. Lassen Sie sich alle Krimis zeigen.

Die Anführungszeichen sagen dem Programm, dass es sich um Text handelt. Zwar sind sie nicht unbedingt nötig, da *Access* versucht, sie selber zu setzen, sobald etwas als Text identifiziert wird, aber es ist sicherer, man gibt sie selber ein.

*Lösung:*

txt_kategorie
tbl_filme
<input checked="" type="checkbox"/>
"Krimi"

**Abd. 4.15 Lösung 1. Aufgabe**

2. Lassen Sie sich alle Filme zeigen, die kürzer sind als 100 Minuten.

*Lösung:*

int_FilmLaenge
tbl_filme
Aufsteigend
<input checked="" type="checkbox"/>
<100

**Abd. 4.16 Lösung 2. Aufgabe**

Sie sehen nichts? Schauen Sie mal nach, ob Sie vorher bei **txt\_kategorie** die andere Bedingung gelöscht haben. Sonst haben Sie nämlich das logische UND in ihre Abfrage eingebaut, was hier noch nicht gefragt war !

3. Lassen Sie sich alle Filme zeigen, die länger sind als 100 Minuten UND Krimis sind.

*Lösung:*

int_FilmLaenge	txt_kategorie
tbl_filme	tbl_filme
Aufsteigend	
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
>100	"Krimi"

**Abd. 4.17 Lösung 3. Aufgabe**

4. Lassen Sie sich alle Filme zeigen, von denen Sie die Kategorie nicht kennen.

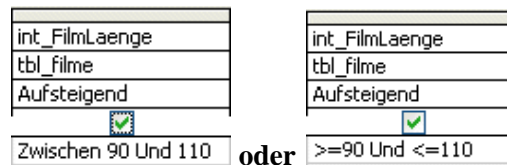
*Lösung:*

txt_kategorie
tbl_filme
<input checked="" type="checkbox"/>
Ist Null

**Abd. 4.18 Lösung 4. Aufgabe**

5. Lassen Sie sich alle Filme zeigen die zwischen 90 und 110 Minuten lang sind.

*Lösung:*



Abd. 4.19 Lösung 5. Aufgabe mit Alternative

Jetzt kommen mal ein paar Beispiele ohne Lösung. Die verschiedenen Möglichkeiten der Abfrage sehen Sie ja oben in den Tabellen.

6. Lassen Sie sich die Filme anzeigen, in deren Titel "Bond" vorkommt. Tip: arbeiten Sie mit den Stellvertreterzeichen (\*).
7. Suchen Sie den Film in dem Madonna oder so ähnlich die weibliche Hauptrolle spielt. Sie wissen allerdings nicht genau, wie die Frau sich schreibt: Madanna, Madonna, oder Madanna.
8. Lassen Sie sich nur Filmtitel und Filmlänge anzeigen.
9. Nehmen Sie die Bandnummer mit dazu. Sie soll an erster Stelle stehen.
10. Lassen Sie sich wieder alle Felder anzeigen.
11. Lassen Sie sich Bandnummer, Hauptdarsteller, Filmlänge und Filmtitel anzeigen und zwar genau in dieser Reihenfolge.
12. Speichern Sie diese Abfrage zur weiteren Verwendung, indem Sie auf den Befehl *Speichern* in der Symbolleiste für den Schnellzugriff klicken. Tragen Sie für die Abfrage den Namen: **"qry\_Kurzform\_mit\_Schauspielern"** ein und bestätigen Sie. Wenn Sie jetzt das Abfragefenster schließen, sehen Sie im Datenbankfenster Ihre neu erstellte Abfrage, die Sie jederzeit neu, weiter oder sonst wie bearbeiten können.
14. Lassen Sie sich auf Grundlage der Abfrage "qry\_Kurzform\_mit\_Schauspielern" nur die Filme, die länger als 90 Minuten sind, anzeigen.
15. Setzen Sie wieder alle Felder in die Abfrage, sortieren Sie sie in erster Instanz nach den Schauspielern, in zweiter Instanz nach der Länge und in dritter Instanz nach der Bandnummer.
16. Sortieren Sie die Tabelle nach den Bändern.
17. Lassen Sie sich auf dieser Grundlage alle Abenteuerfilme und Bondfilme anzeigen. Speichern Sie das nicht.

Das sollte zu diesem Thema reichen. Wie gesagt: die Formeln können beliebig schwierig werden, je nachdem, welche Bedingungen man für sein Dynaset braucht.

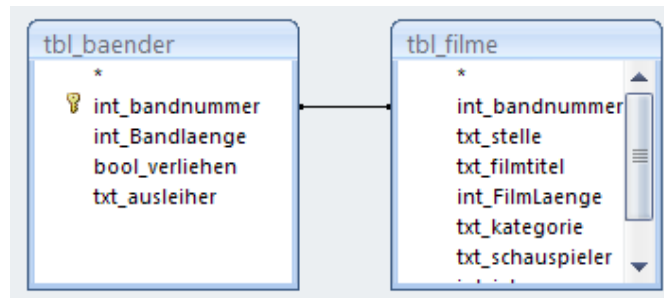
### 4.3.4 Komplexes

#### 4.3.4.1 Verknüpfen von Tabellen in Abfragen

Es wurde schon gezeigt, dass man in *Access* mehrere Tabellen verknüpfen kann. In Abfragen kann man diese verknüpften Tabellen endlich benutzen. Man kann hier die Erklärung gleich mit einer Übung verbinden.

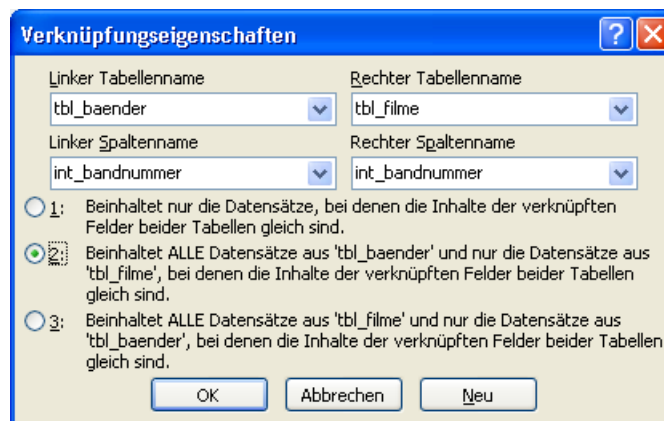
## Übung

1. Erstellen Sie eine neue Abfrage in der Entwurfsansicht. Das sich öffnende Bild kennen Sie schon. Klicken Sie **tbl\_filme** an und drücken Sie auf **Hinzufügen**. Danach markieren Sie **tbl\_baender** und betätigen **Hinzufügen**.
2. Sie sehen, dass das Programm die Verknüpfung zwischen den Spalten *Bandnummer* beider Tabellen übernommen hat, die man vorher unter **Beziehungen** eingerichtet hat. Übrigens: Falls das nicht der Fall sein sollte, klickt man einfach mit der Maus auf den Spaltennamen *int\_bandnummer* der einen Tabelle und zieht die Überschrift mit gedrückter Maustaste zur Spalte *int\_bandnummer* der anderen Tabelle. Das Ergebnis sieht dann so aus:



Abd. 4.20 Verknüpfung zwischen zwei Tabellen in einer Abfrage

Auch hier müssen die die Felddatentypen der zu verknüpfenden Felder gleich sein, damit **Access** die Verknüpfung akzeptiert. Allerdings kann man keine Angaben zur referentiellen Integrität machen, weshalb die Verknüpfung über den Befehl **Extras/Beziehungen** eindeutig professioneller ist. Ein Doppelklick auf den Verbindungsstrich erlaubt hier auch die Angabe des Verknüpfungstyps




Abd. 4.21 Verknüpfungseigenschaften in Abfragen

3. Erstellen Sie durch herunterziehen der richtigen Felder in der richtigen Reihenfolge die auf dem Bild sichtbare Abfrage:

Feld:	txt_filmtitel	int_jahr	txt_schauspieler	int_FilmLaenge	int_bandnummer	int_Bandlaenge
Tabelle:	tbl_filme	tbl_filme	tbl_filme	tbl_filme	tbl_filme	tbl_baender
Sortierung:						
Anzeigen:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kriterien:						
oder:						

Abd.4.22 Abfrage "Filme und Bänder verbunden"

4. Speichern Sie sie unter dem Namen "qry\_FilmeBaender" und schauen sich die Auswahlliste mit  an. Wenn alles gut geklappt hat, sollte folgender Bildschirm bei Ihnen zu sehen sein:

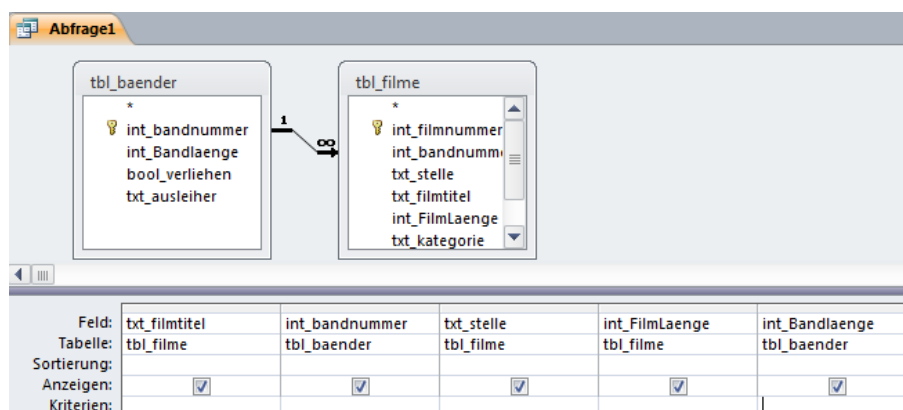
Filmtitel	Jahr	Schauspieler	Länge	Bandnummer	Bandlänge
Buck Rogers	0		85 min	1	180
Krieg der Epiraten	0		90 min	1	180
Scotland Yard jagt Dr. Mabuse	0		90 min	2	180
Auf der Jagd nach dem Juwel vom Nil	0	Michael Douglas	110 min	2	180
Stirb langsam II	0	Bruce Willis	140 min	2	180
Auf der Jagd nach dem grünen Diamanten	0	Michael Douglas	120 min	3	240
Die Spezialisten	0		90 min	3	240
Red Heat	0	Arnold Schwarzenegger, James Belushi	83 min	3	240
Loriot V	0	Loriot	45 min	4	180
Loriot VI	0	Loriot	45 min	4	180
Tanz der Vampire	0	Roman Polanski	110 min	4	180
Liebesgrüße aus Moskau	0		120 min	5	240
Drei Männer im Schnee	0	Gustav Knuth	90 min	5	240
Werbespots	0		240 min	6	240
Loriot I	0	Loriot	45 min	7	180

Abd.4.23 Abfrageergebnis von "qry\_Filme\_und\_Baender verbunden"

#### 4.3.4.2 Berechnen von neuen Feldern

Schaut man sich unser Dynaset so an, ist das alles schon sehr schön, aber noch schöner wäre es, wenn wir uns auch die freie Zeit auf den Bändern anzeigen lassen könnten. Dem wollen wir uns jetzt nähern.

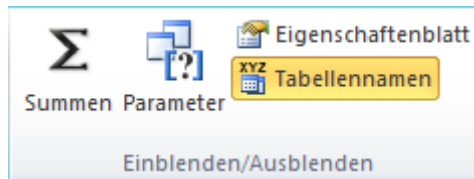
Die erste Überlegung ist natürlich wieder, welche Felder wir brauchen, um eine Abfrage zu erstellen, die dieses Problem bewältigt. Zum einen benötigen wir die **Filmtitel**, **Filmlänge** und die **Stelle der Filme**, an denen sie aufgenommen worden sind. Dann brauchen wir die **Bandnummer** und die **Gesamtlänge der Bänder**. Wenn soweit Einigkeit herrscht, dann versuchen Sie einmal, folgende Abfrage zu erstellen, die Sie unter "qry\_Freie\_Bandzeit" speichern.



Abd.4.24 Auswahlfrage: Freie Bandzeiten

**Achtung:** Drei Sachen gilt es zu bedenken: Erstens muss es eine Verknüpfung sein, bei der **alle** Bänder gezeigt werden, damit wir auch die ganz leeren bekommen (Verknüpfung 2). Zweitens muss man deshalb das Feld **Bandnummer** aus der zweiten Tabelle (Bänder) nehmen, sonst bleibt der Eintrag nämlich leer. Drittens sollte man deshalb auch die Tabellennamen mit einblenden, damit man weiß, welches Feld aus welcher Tabelle stammt. Normaler-

weise erscheint die Zeile Tabelle automatisch, ist also Standardeinstellung. Wenn nicht machen macht man folgendes: Man wählt den Befehl **Tabellennamen** aus der Multifunktionsleiste.



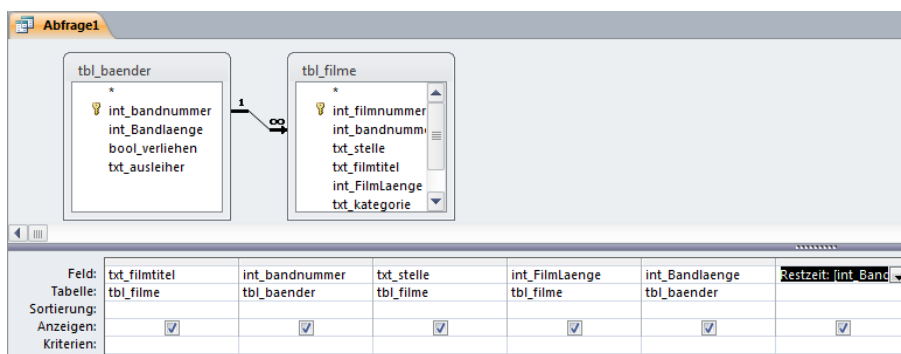
Abd.4.25 1. Tabellennamen in Abfragen ein- und ausblenden

Über der Zeile "Sortierung" bekommen Sie die Zeile "Tabelle". Probieren Sie es ruhig einmal aus. Das Dynaset sollte dann ungefähr so aussehen:

qry_freie_Bandzeit	Filmtitel	Bandnummer	Stelle auf dem Band	Länge	Bandlänge
	Buck Rogers	1	a	85 min	180
	Krieg der Epiraten	1	b	90 min	180
	Scotland Yard jagt Dr. Mabuse	2	a	90 min	180
	Auf der Jagd nach dem Juwel vom Nil	2	b	110 min	180
	Stirb langsam II	2	c	140 min	180
	Auf der Jagd nach dem grünen Diamanten	3	a	120 min	240
	Die Spezialisten	3	b	90 min	240
	Red Heat	3	c	83 min	240
	Loriot V	4	a	45 min	180
	Loriot VI	4	b	45 min	180
	Tanz der Vampire	4	c	110 min	180
	Liebesgrüße aus Moskau	5	a	120 min	240
	Drei Männer im Schnee	5	b	90 min	240
	Werbespots	6	a	240 min	240

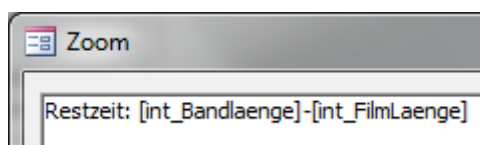
Abd.4.26 1. Schritt: Dynaset zur freien Bandzeit

Schon ganz schön, aber hier müssten wir noch mit der Hand rechnen. Also werden wir jetzt ein Feld einfügen, das diese Arbeit für uns übernimmt. Wechseln Sie wieder in die Entwurfsansicht und tragen Sie von Hand eine weitere Spalte ein. Die neue Spalte erstellen wir an der ersten freien Stelle, also rechts neben die Spalte **Bandlänge**.



Abd.4.27 1. Ein neues, berechnetes Feld erstellen

Stellen Sie den Cursor in die Zeile **Feld** der neuen Spalte, drücken Sie **Umschalten + F2**, damit der Zoom eingeblendet wird und schreiben Sie die folgende Formel dorthin:



Abd. 4.28 Eintragen der Formel



Die Logik bei diesen Rechenformeln ist immer gleich. Sie schreiben erst einen von Ihnen erfundenen Feldnamen, der mit einem Doppelpunkt abgeschlossen wird - hier: **Restzeit**: -, dann folgen in **eckigen** Klammern die Feldnamen, mit denen Sie rechnen wollen. Dazwischen sind alle Rechenoperationen erlaubt, die weiter vorne schon einmal angeführt wurden.

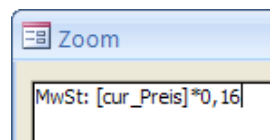
Schaut man sich jetzt sein Dynaset an, sollte es ungefähr so aussehen:

Filmtitel	Bandnummer	Stelle auf dem Band	Länge	Bandlänge	Restzeit
Buck Rogers	1	a	85 min	180	95
Krieg der Eispiraten	1	b	90 min	180	90
Scotland Yard jagt Dr. Mabuse	2	a	90 min	180	90
Auf der Jagd nach dem Juwel vom Nil	2	b	110 min	180	70
Stirb langsam II	2	c	140 min	180	40
Auf der Jagd nach dem grünen Diamanten	3	a	120 min	240	120
Die Spezialisten	3	b	90 min	240	150
Red Heat	3	c	83 min	240	157
Loriot V	4	a	45 min	180	135
Loriot VI	4	b	45 min	180	135
Tanz der Vampire	4	c	110 min	180	70
Liebesgrüße aus Moskau	5	a	120 min	240	120
Drei Männer im Schnee	5	b	90 min	240	150
Werbespots	6	a	240 min	240	0
Loriot I	7	a	45 min	180	135
Loriot II	7	b	45 min	180	135

Abd.4.29 3. Schritt: Dynaset mit Restzeit

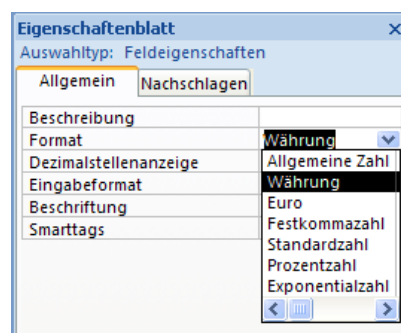
Es wird immer besser, aber zum einen merkt die Abfrage nicht, dass auf einem Band eventuell zwei Filme sind, zum anderen werden bei leeren Bändern keine Berechnungen angestellt. Damit kann man leben, aber es geht noch besser. Zum einen mit Abfragefunktionen, die wir in diesem Kurs jedoch nicht mehr behandeln, zum anderen wenn man die letzte Kategorie, die für diesen Kurs vorgesehen ist, zur Hilfe nimmt, nämlich den Bericht (siehe 6. Kapitel).

**Ein anderes Beispiel:** Man arbeitet in einer Artikelliste mit Preisen. Die Mehrwertsteuer kann man sich in der Abfrage ausrechnen lassen:



Abd. 4.30 Mögliche Formel für die Mehrwertsteuer

Leider kennt **Access** an dieser Stelle das Rechenzeichen % nicht an. Man muss mit 0,16 bzw. mit 16/100 rechnen. Das gleich gehe übrigens auch später in den Formularfeldern und die selben Regeln gelten in Kapitel 6 bei den Berichten. Möchte man übrigens diese Spalte auch formatiert haben als Währung, z. B. €, so macht man das mit der rechten Maustaste im Abfragefenster. In der entsprechenden Spalte wählt man den Befehl **Eigenschaften**. Es öffnet sich das Eigenschaftenblatt auf der rechten Bildschirmseite.



Abd. 4.31 Eigenschaften gibt es auch im Abfragefenster

Um es perfekt zu machen könnte man noch ein weiteres Feld berechnen lassen, nämlich den Endpreis mit Mehrwertsteuer. Das Ergebnis könnte dann so aussehen:

ID	Artikelbezeichnung	Preis (ohne MwSt)	MwSt	Preis mit MwSt
2	Schlafsack	139,60 €	22,34 €	161,94 €
3	Campingkocher	24,80 €	3,97 €	28,77 €
4	Taschenlampe	48,90 €	7,82 €	56,72 €
* (Neu)				

Abd. 4.32 Dynaset mit 2 berechneten Feldern

#### 4.3.4.3 Abfrage mit Parametern

Abfragen sind so flexibel, dass man statt des festen Wertes, den **Access** heraus suchen soll, auch einen Stellvertreter als Kriterium vorsehen kann, so dass zur Ablaufzeit der Abfrage nachgehakt wird, was man denn eigentlich braucht.

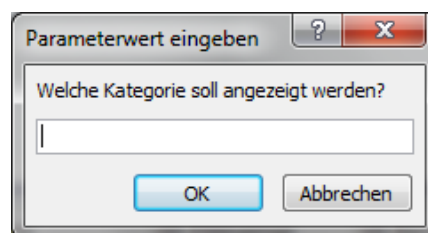
Im Folgenden wird dies anhand der Spalte *Kategorie* erläutert. Dazu wird die Spalte *Kategorie* aus dem Auswahlbereich in die Entwurfsansicht gezogen. In die Zeile Kriterien schreibt man folgendes hinein:

[Welche Kategorie soll angezeigt werden?]

Feld:	txt_kategorie
Tabelle:	tbl_filme
Sortierung:	
Anzeigen:	<input checked="" type="checkbox"/>
Kriterien:	[Welche Kategorie soll
oder:	

Abd. 4.33 Parameter

Der Trick besteht darin, dass man die Frage in eckige Klammern setzt. Immer wenn das Programm eine Aussage in der Zeile Kriterien in diesen Klammern findet, tut es so, als wäre es gefragt. Die Antwortmöglichkeit erscheint wenn man nun die Abfrage laufen lässt.



Abd. 4.34 Parameterwert eingeben

Je nachdem welche Kategorie gewünscht ist, gibt man sie dort ein. Die Eingabe von „Western“ zeigt alle Westernfilme an, wenn man vorher noch auf **OK** klickt. Folgendes Bild erscheint:

Kategorie	Filmtitle	Bandnummer	Stelle auf dem Band	Länge	Bandlänge	Restzeit
Western	Der Mann, der die Katzen tanzen läßt	10 b		110 min	240	130
Western	Der mit dem Wolf tanzt	73 a		255 min	240	-15
* (Neu)						

Abd. 4.35 Dynaset nach Parametereingabe

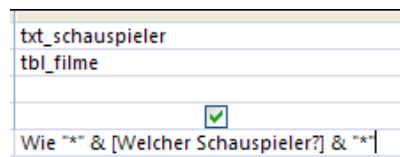
## Übung

Probieren Sie es einmal selber mit der Suche nach einer bestimmten Bandnummer (z.B. Welche Bandnummer suchen Sie?).

### 4.3.4.4 ... und Stellvertreterzeichen

Perfekt wird eine solche Parameterabfrage, wenn man sie noch mit Stellvertreterzeichen mischen kann. Leider geht es nicht so, dass man einfach in das sich öffnende Fenster eingibt: *\*Bond\**

Damit bekommt man kein Ergebnis. Die Stellvertreterzeichen müssen bereits in der Parameterabfrage enthalten sein. Das sieht dann so aus:



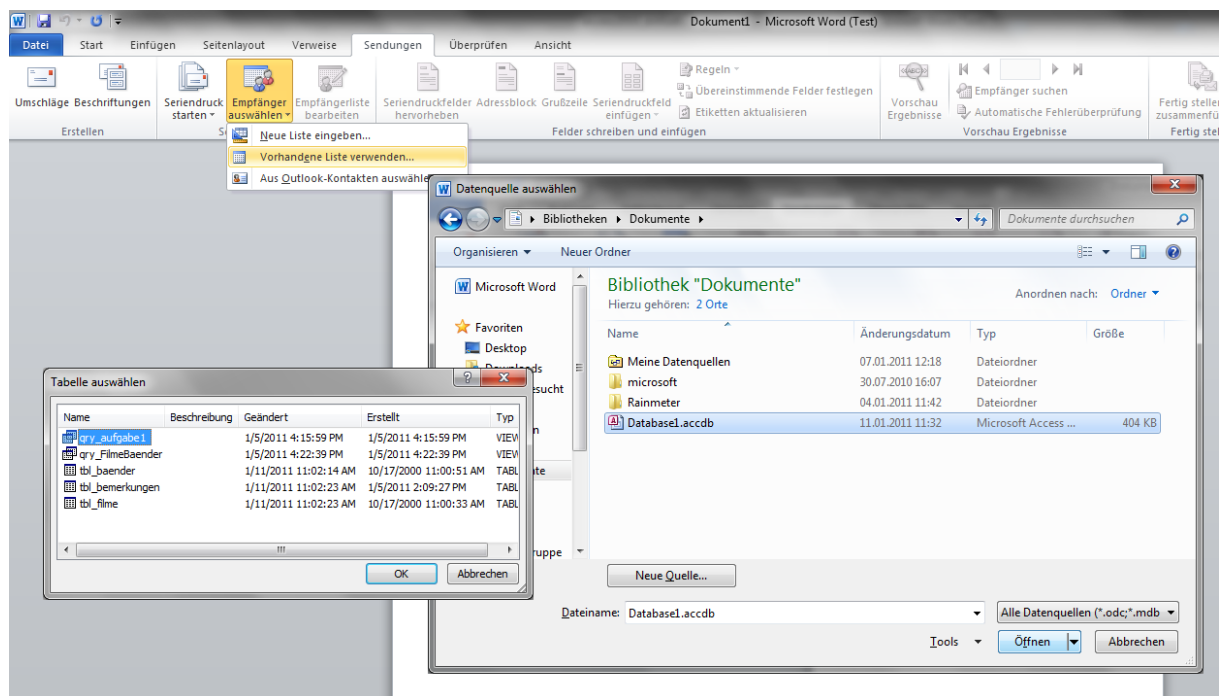
Abd. 4.36 Parameterabfrage mit Stellvertreterzeichen

## 4.4 Serienbriefe aus Access-Daten erstellen

Serienbriefe mit dem Programm Word können sehr schnell und einfach mit den Daten einer Auswahlabfrage oder einer bestimmten Tabelle verbunden werden.

Hierzu muss lediglich bei der Auswahl der Datenquelle die richtige Access-Datei angegeben werden. Daraufhin kann aus sämtlichen Abfragen und Tabellen der Access-Datei eine Abfrage oder Tabelle als Datenquelle ausgewählt werden.

Es empfiehlt sich daher, im Vorfeld eine Abfrage für den Seriendruck zu erstellen, die genau die benötigten Daten findet.



Abd. 4.37 Auswahl einer Access-Datei als Datenquelle für einen Seriendruck in MS Word 2010

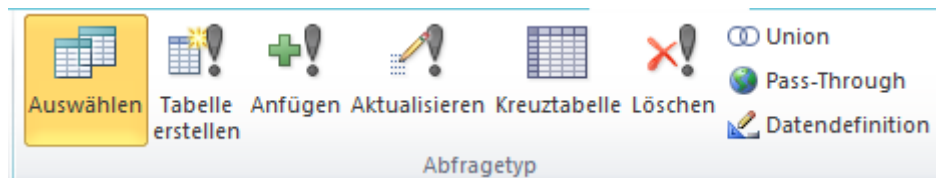
## 4.5 Andere Abfragen

Bisher wurden Abfragen immer erstellt als selektiertes Abbild der Tabellen oder anderer Abfragen. Diese heißen in *Access* „Auswahlabfragen“. Schon die Tatsache, dass ein eigener Name vorliegt, lässt darauf schließen, dass dort noch anderes zu erwarten ist. Und so ist es auch. Es existiert noch ein ganzer Schwung von Abfragen, die sich „Ereignisabfragen“ nennen. Deren Gemeinsamkeit besteht darin, dass man ein Ergebnis in einer neuen oder geänderten Tabelle bekommt.

### 4.5.1 Die Tabellenerstellungsabfrage

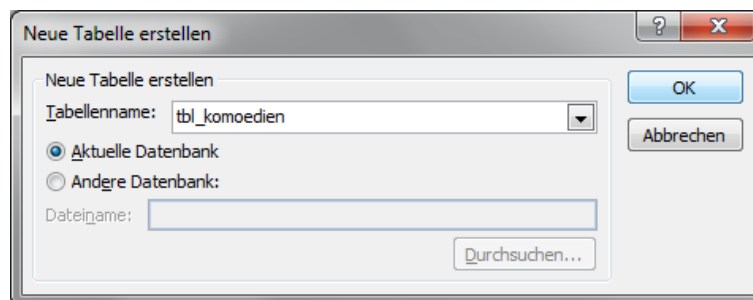
Wenn man zwecks Export von Daten eine neue Tabelle braucht, in der nur bestimmte Datensätze gespeichert sind, dann reicht eine Auswahlabfrage nicht, da diese ja nur immer wieder den Befehl ausführt, aber nicht wirklich eine physische neue Tabelle erstellt. Um dies zu erreichen, braucht man die „Tabellenerstellungsabfrage“.

Man findet diese Funktionalität, wenn man eine neue Abfrage erstellt und dann in der Multifunktionsleiste auf der Registerkarte **Entwurf** auf den Befehl **Tabelle erstellen** geht.



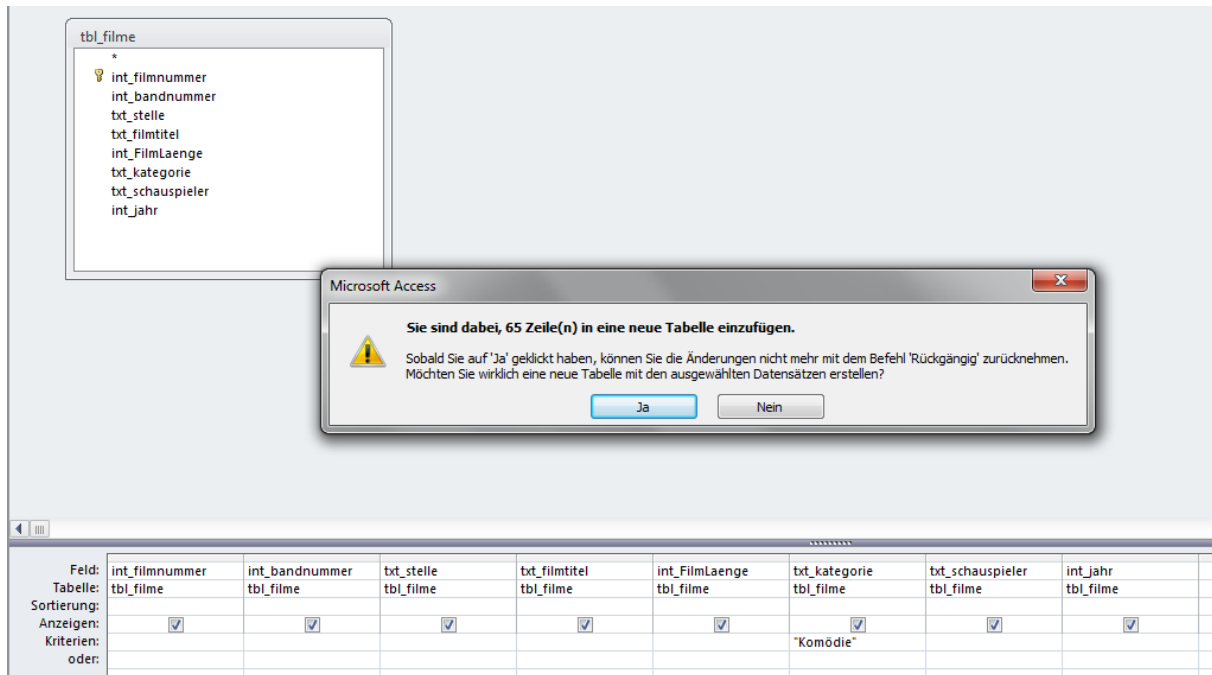
Abd. 4.38 Die verschiedenen Abfragetypen

Statt dem standardmäßig markierten „Auswählen“ wählt man einfach „Tabelle erstellen“. *Access* zeigt einem umgehend ein neues Fenster, in das man den Namen der neuen Tabelle eintragen soll:



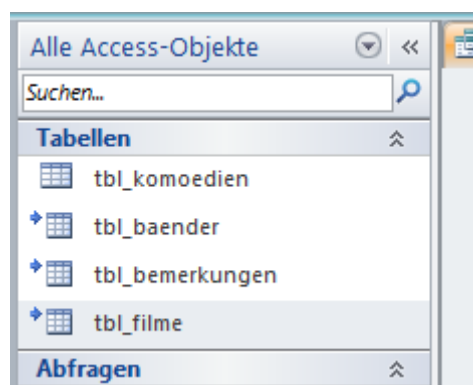
Abd. 4.39 Die Tabellenerstellungsabfrage braucht einen Namen für die neue Tabelle

Kriterien und Spalten werden wie gewohnt definiert und danach klickt man auf das wohlbekannte rote Ausrufezeichen, um die Abfrage zum Laufen zu bringen. An der Stelle kommt eine weitere Nachfrage.



Abd. 4.40 Nachfragen von Access

Ist man sicher, dass man die neue Tabelle, in der in diesem Beispiel dann nur die Komödien stehen, wirklich erstellen möchte, bestätigt man es mit **Ja** und findet Sekunden später im Navigationsbereich auf der linken Seite die neue Tabelle vor.



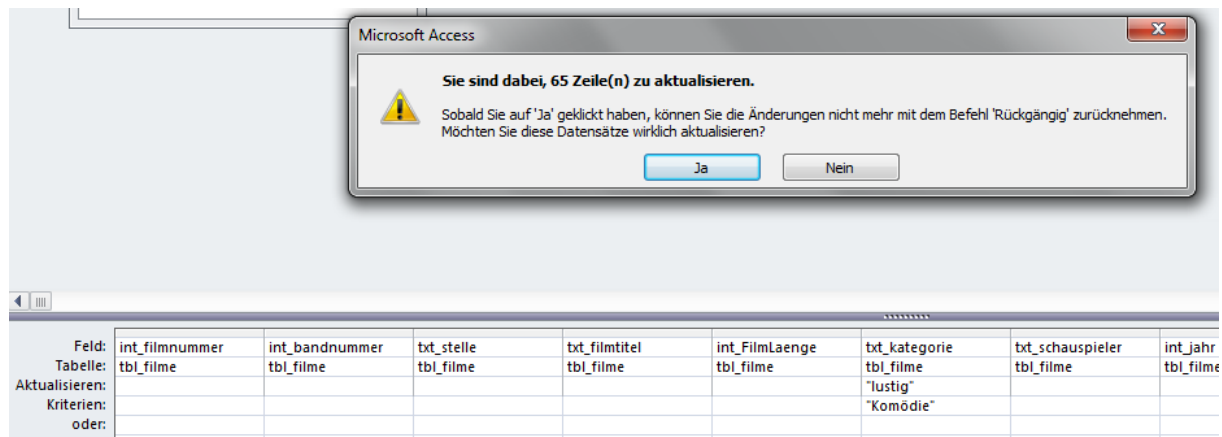
Abd. 4.41 Die neue Tabelle im Register Tabellen

Schwierig ist das nicht. Diese Tabelle kann man dann weitergeben, da sie alle Daten enthält, die den Kriterien genügen.

#### 4.5.2 Die Aktualisierungsabfrage

Eine andere Art der Ereignisabfrage ist die „Aktualisierungsabfrage“. Mit ihr kann man in einer Tabelle auf einen Rutsch einen Begriff austauschen. Sollen z. B. die Filme, die mit der Kategorie *Komödie* belegt wurden, in *lustig* untergebracht werden, erstellt man wieder eine neue Abfrage, holt sich die Spalte, in der der zu ändernde Begriff steht und wählt diesmal „Aktualisieren“. Jetzt wird nicht nach einem neuen Dateinamen gefragt, sondern unten der Abfragebereich ändert sich.

Statt der Zeile *Sortieren* finden wir dort *Aktualisieren*. Dort hinein schreibt man den neuen Begriff, also in diesem Beispiel „lustig“. In die *Kriterien* schreibt man das, wonach *Access* suchen soll, nämlich „Komödie“. Lässt man die Abfrage mit dem roten Ausrufezeichen laufen, erscheint wieder eine Nachfrage:

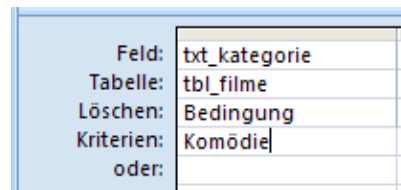


Abd. 4.42 Datensätze werden aktualisiert

Das ist auch gut so, da man in dem Moment in der originalen Tabelle Änderungen durchführt. Die Bestätigung mit *Ja* ändert in der Tabelle Filme im Feld Kategorie überall „Komödie“ in „lustig“.

### 4.5.3 Die Löscharfrage

Nicht ganz ungefährlich ist die „Löscharfrage“, da sie alle Datensätze der Tabelle löscht, die dem angegebenen Kriterium genügen. Hat man die Schnauze voll von all den Komödien, könnte man eine neue Abfrage erstellen, dort das Feld „txt\_Kategorie“ nach unten holen und „Löschen“ wählen.



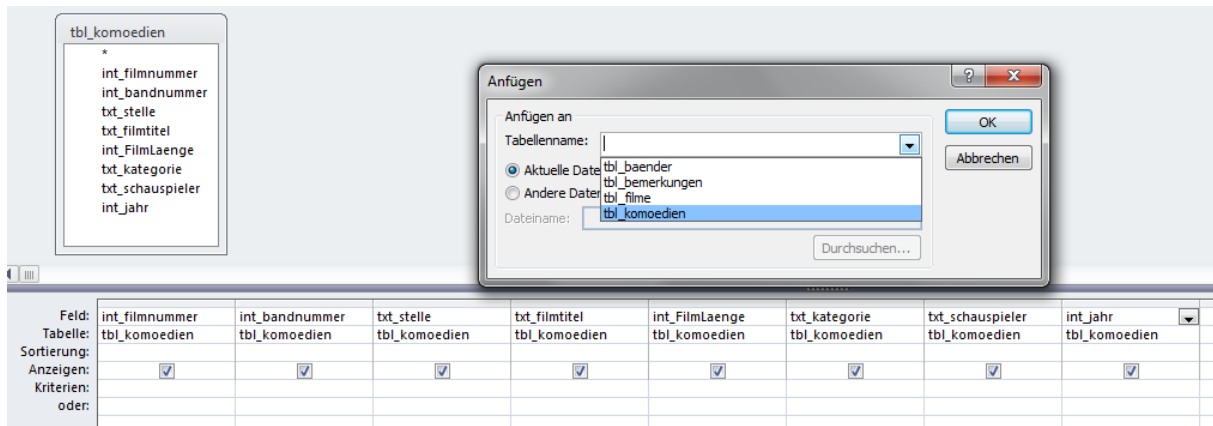
Abd. 4.43 Alle Komödien werden gelöscht

Nach Klicken auf das rote Fragezeichen auch hier wieder die schon bekannte Nachfrage. Schaut man nach der Bestätigung in seine Filmtabelle, sind alle Komödien weg. Schade.

### 4.5.4 Die Anfügeabfrage

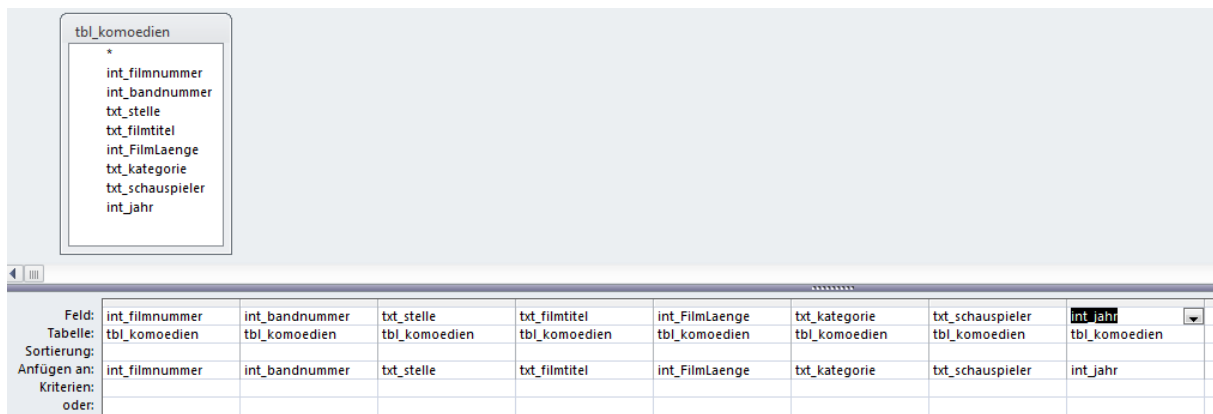
Die letzte Aktions- oder Ereignisabfrage ermöglicht es, Daten aus einer Tabelle an eine zweite anzufügen. Das ist ganz praktisch, wenn man Daten von einem anderen System bekommt, die in eigene Tabellen integriert werden müssen. Auch hier erstellt man eine neue Abfrage und wählt dann als Tabelle diejenige aus, die man anfügen möchte.

Für unser Beispiel werden die Daten aus tbl\_komoedien genommen und an tbl\_filme angefügt, damit diese Filme nach dem Löschen im vorigen Schritt wieder in der Haupttabelle hergestellt werden. Sobald man in der Multifunktionsleiste „Anfügen“ anklickt, erscheint ein Fenster zur Auswahl der Tabelle, an die die Daten angefügt werden sollen.



Abd. 4.44 Die tbl\_komoedien soll an die tbl\_filme angefügt werden

Hat man die entsprechende Tabelle ausgewählt, ändert sich wiederum der Abfragebereich:



Abd. 4.45 Welche Spalten sollen wo angefügt werden?

Hier kann man jetzt angeben, welche Feldnamen der ersten Tabelle, den Feldnamen der zweiten Tabelle entsprechen. Die Namen müssen also nicht gleich sein, aber die Felddatentypen sollten es schon sein, da sonst Datenverlust zu befürchten ist. Hat man alle Spalten angepasst, erzeugt der Klick auf das rote Fragezeichen wieder eine Nachfrage.

Danach finden sich in diesem Beispiel alle Komödien wieder in der Haupttabelle.

Soweit zu den verschiedenen Abfragetypen, die sich als mächtiges Mittel der Datenpflege erweisen.



## 5 Formulare erstellen

Im 3. Kapitel wurde bereits erklärt, wie man eine Tabelle erstellt und mit ihr arbeitet. Um das Bearbeiten und Ansehen der Daten bequemer und freundlicher zu gestalten, gibt es in **Access** die Möglichkeit, sich selber Formulare zur erstellen. In der Regel wird man die Benutzer einer Datenbank nur sehr selten mit den reinen Tabellen oder Abfragen arbeiten lassen sondern den Datenzugriff komplett über Formulare und Berichte realisieren. Wie man ein Formular erstellt und benutzt wird in diesem Kapitel erläutert.

In 99 % der Fälle beruht ein Formular auf einer Tabelle oder Abfrage. Alle Feldnamen kommen aus einem Tabellenentwurf. Man kann jedoch frei bestimmen in welcher Reihenfolge diese Felder auf dem Bildschirm erscheinen. Für unser Datenbank-Beispiel **Videothek2010FE.accdb** wird im Folgenden ein Formular für das Bearbeiten der Tabelle **tbl\_filme** erstellt.

Es gibt mehrere Möglichkeiten, ein Formular zu erstellen. Mit dem Formularassistenten und den Auto-Formularen wird die Sache sehr einfach. Die dritte Möglichkeit ist es, sich selber nach eigenen Wünschen und Vorstellungen ein Formular zu erstellen. Diese Methode ist aufwendiger und wird in diesem Skript nicht besprochen. Im Rahmen dieses Kapitels wird die zweite Möglichkeit behandelt, die Auto-Formulare. Das soll einen aber nicht davon abhalten, es auch einmal mit dem Assistenten auszuprobieren. *Danach kann man "von Hand" noch Änderungen vornehmen.*

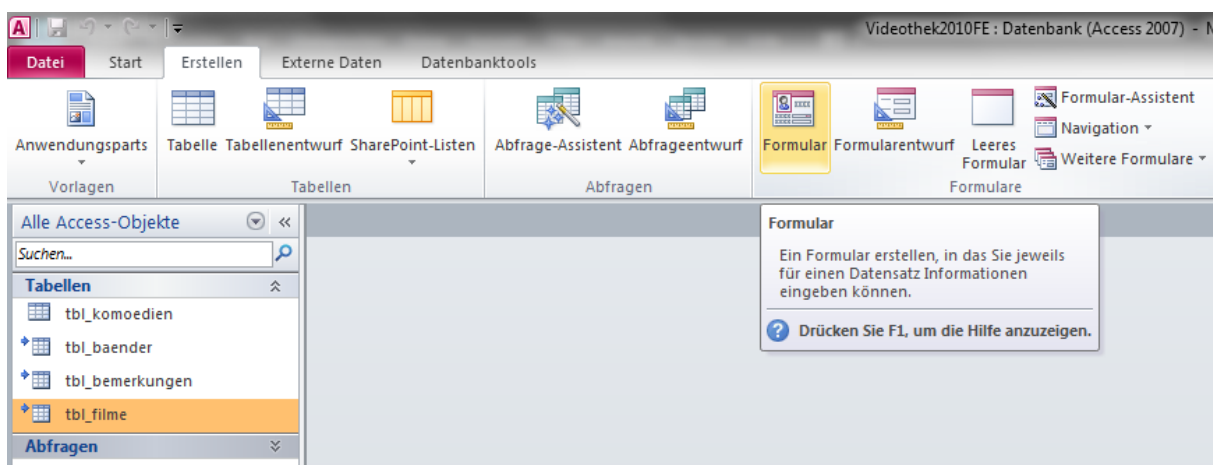
### 5.1 Formulare mit dem Auto-Formular erstellen

Wie schon erwähnt, ist die Erstellung von Formularen mit Hilfe der Auto-Formulare sehr einfach und schnell. Wie funktioniert es?

#### Übung

Vorgehensweise bei dem Erstellen eines Formulars mit dem Auto-Formular:

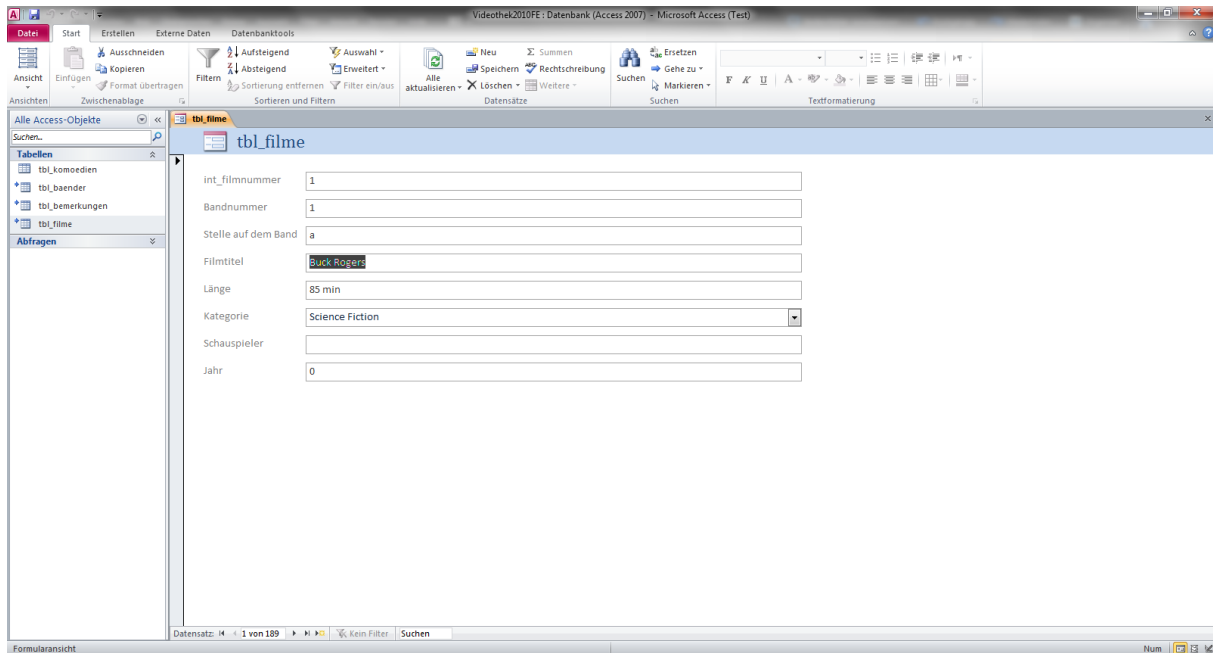
1. Markieren Sie durch einen einfachen Mausklick im Navigationsbereich die Tabelle oder Abfrage, die als Grundlage für Ihr Formular dienen soll. In unserem Fall ist dies die Tabelle **tbl\_filme**.
2. Klicken Sie in der Multifunktionsleiste auf der Registerkarte **Erstellen** auf den Befehl **Formular**.



Abd. 5.1 Registerkarte Erstellen, Befehlsgruppe Formulare

3. Jetzt erscheint schon ihr fertiges Formular (s. Abd. 5.2). Einfach und schnell.





**Abd. 5.2 Fertiges Formular in der Formularansicht**


Die verschiedenen Befehle für Auto-Formulare haben die folgende Bedeutung:

- Formular:** Jeder Datensatz in der Tabelle stellt eine Zeile dar, im Formular erscheint jeder Datensatz auf einer eigenen "Seite". Das sieht einer Karteikarte am ähnlichsten
- Geteiltes Formular:** Im oberen Teil erscheint ein Formular und im unteren Teil erscheint die Datenblattansicht der Tabelle.
- Mehrere Elemente:** Alle Datenfelder eines Datensatzes stehen von links nach rechts in einer Zeile. Dieses Formular ähnelt der Datenblattansicht einer Tabelle.
- Datenblatt:** Erstellt das Formular in der Datenblattansicht, in der man Daten bearbeiten kann.
- Formular-Assistent:** Startet den Formular-Assistenten.
- PivotTable* und *PivotChart* können wir ebenso wie *Modales Dialogfeld* getrost vergessen.

Ein Auto-Formular zu verwenden bedeutet jedoch nicht, dass man keine Änderungen am Layout durchführen könnte. Das wird erst später besprochen. Jetzt beschäftigen wir uns damit, wie man mit diesem Formular arbeitet.

## 5.2 Arbeiten mit Formularen

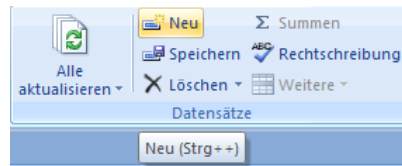
### 5.2.1 Bewegen in Formularen

Wie man bereits bei der Arbeit mit Tabellen kennen gelernt hat, gibt es auch bei Formularen die so genannten Navigationssymbole, um bestimmte Datensätze zu erreichen. Erinnern Sie sich noch an diese Symbole?  (Kapitel 3)


## 5.2.2 Datensätze hinzufügen

Es bestehen vier Möglichkeiten, einen neuen Datensatz in ein Formular einzugeben, und zwar:

1. Wählen Sie aus der Multifunktionsleiste auf der Registerkarte **Start** in der Befehlsgruppe **Datensätze** den Befehl *Neu* oder



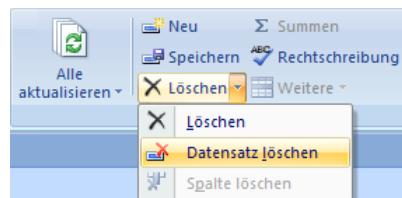
Abd. 5.3 Registerkarte Start, Befehlsgruppe Datensätze

2. drücken sie die Tastenkombination **STRG + Pluszeichen** oder
3. klicken sie in der Navigationsleiste des Formulars auf diesen Button  oder
4. gehen Sie zum letzten Feld des letzten Datensatzes und drücken die TAB-Taste.

Bei den oben genannten Möglichkeiten zeigt **Access** danach einen leeren Datensatz an. Geben Sie die gewünschten Daten in den Datensatz ein. Mit [Tab], [↓], [↑], [Returntaste] können Sie von einem Datenfeld bzw. Datensatz zum nächsten springen. Zum ersten Feld im Formular kommen Sie mit der Taste [Pos1] und zum letzten Feld mit [Ende].

## 5.2.3 Datensätze löschen

Um einen Datensatz zu löschen, geht man zu dem Datensatz und wählt aus der Multifunktionsleiste auf der Registerkarte **Start** den Befehl *Datensatz löschen*.




Abd. 5.4 Datensatz löschen

**Access** fordert den Benutzer auf, den Löschvorgang zu bestätigen. Man wählt **OK**, um den markierten Satz zu löschen, oder **Abbrechen**, um den Löschvorgang rückgängig zu machen. Wie man sieht, funktioniert es genauso wie bei einer Tabelle.

## 5.2.4 Speicherung von Eingaben und Änderungen

**Access** speichert neu eingegebene Datensätze und Änderungen automatisch, wenn man zum nächsten oder vorherigen Datensatz springt. Eine andere Möglichkeit wäre es, den Befehl *Speichern* (Umschalt + Eingabe) aus der Befehlsgruppe **Datensätze** zu wählen.


## 5.2.5 Änderungen rückgängig machen

Mit der Taste [Esc] oder dem Symbol  kann man eine Neueingabe abbrechen oder eine Änderung des Datensatzes rückgängig machen. Allerdings funktioniert der Trick mit der Taste [Esc] nur solange man sich noch in dem Datensatz befindet, er - siehe oben - also noch nicht gespeichert ist. Mit *Rückgängig* kann man nur den letzten Befehl wieder aufheben, also ist das auch nicht der Renner.

## Übung

1. Geben sie einen neuen Film (z.B. ihren Lieblingsfilm) über das Formular in die Datenbank ein.
2. Nachdem Sie den neuen Datensatz eingegeben haben, können Sie über die Multifunktionsleiste die **Datenblattansicht** wählen und sich dort den hinzugefügten Datensatz anschauen. Die Datenblattansicht eines Formulars sieht genauso aus, wie die der Tabelle.
3. Löschen sie den neuen Datensatz wieder. Gemein, oder ?

### 5.2.6 Suchen und Ersetzen

Ähnlich wie bei einer Tabelle gibt es beim Formular auch die Möglichkeit; Datensätze nach bestimmten Kriterien zu suchen. Sowohl in der Datenblattansicht als auch in der Formularansicht funktioniert das Suchen. Mit dem Befehl *Suchen* aus der Navigationsleiste oder dem Symbol  aus der Multifunktionsleiste kann man sofort mit der Suche beginnen. (Genauer, siehe Kap. 3.0)

### Übung

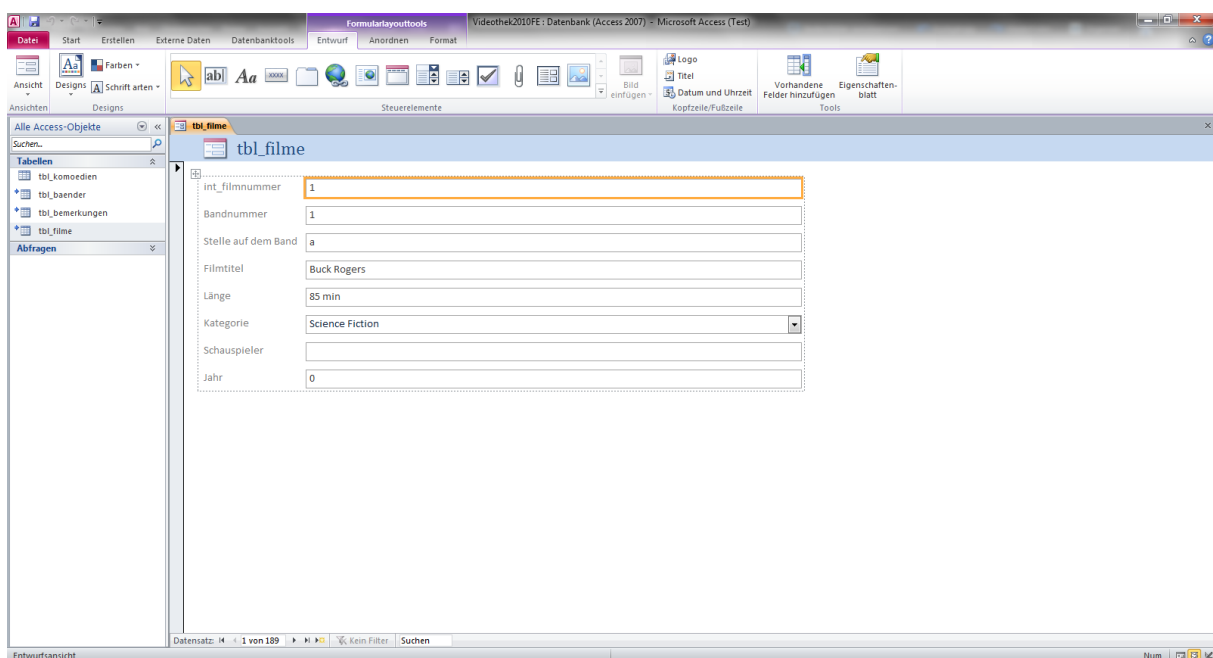
Suchen Sie alle Filme mit dem Genre "Komödie".

## 5.3 Ändern eines Formularlayouts

Wie schon angekündigt, kann man das Formularlayout nach seinen eigenen Vorstellungen ändern. Das soll Thema der folgenden Punkte sein.

### 5.3.1 Formular in der Layoutansicht

Seit Access 2007 ist es mit der Layoutansicht möglich, Entwurfsänderungen während der Arbeit mit einem Formular oder Bericht vorzunehmen. Sie können also mit dem Formular durch die Datensätze Ihrer Tabelle navigieren und währenddessen das Layout des Formulars ändern. Es können Felder hinzugefügt oder entfernt werden und die Layoutansicht unterstützt ebenfalls die neuen gestapelten und tabellarischen Layouts. Mit Hilfe dieser Layouts können Felder, Spalten und Zellen sehr einfach angepasst werden.

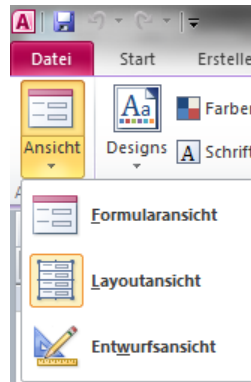


Abd. 5.5 Formular Filme in der Layoutansicht

Die Layoutansicht ist nach der Erstellung eines Auto-Formulars standardmäßig bereits eingestellt. Damit einem alle Möglichkeiten zur Entwurfsänderung offen stehen, muss man jedoch in die Entwurfsansicht wechseln.

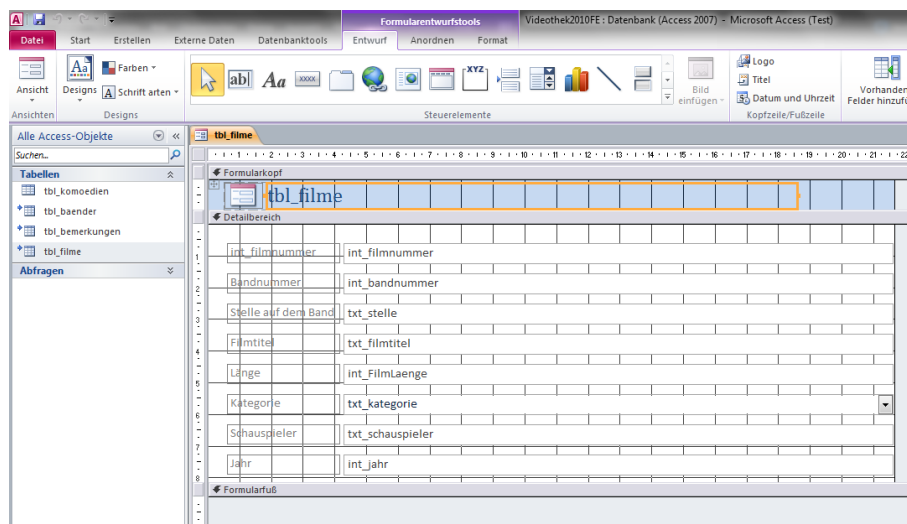
### 5.3.2 Formular in der Entwurfsansicht

Nun muss man logischerweise das Formular Filme in der Entwurfsansicht öffnen. Dazu wählt man aus der Multifunktionsleiste das entsprechende Symbol.



Abd. 5.6 Ansicht eines Formulars wechseln


Eine andere Möglichkeit ist, im Navigationsbereich das Formular Filme auszuwählen und nach einem Rechtsklick den Befehl *Entwurfsansicht* auszuwählen.



Abd. 5.7 Formular Filme in der Entwurfsansicht

So sieht also das Formular **Filme** in der Entwurfsansicht aus. Das Formularentwurf fenster ist in drei Bereiche aufgeteilt:

1. In den *Formularkopf* wird die Überschrift des Formulars automatisch eingetragen und muss u.U. noch angepasst werden. Wie das funktioniert, folgt in Punkt 5.3.3.
2. Der *Detailbereich* enthält alle Felder der **Filme**-Tabelle. Hier stehen rechts die Feldüberschriften und links die Feldinhalte.
3. Der *Formularfuß* enthält Informationen, die am Ende des Formulars erscheinen sollen. Er ist gegenwärtig leer.

Den einzelnen Bereich kann man mit der Maus vergrößern oder verkleinern. Positionieren Sie den Mauszeiger dazu auf dem unteren Rand des entsprechenden Bereichs, so dass der Mauszeiger seine Form zu der eines Doppelpfeils  ändert. Ziehen Sie den veränderten Cursor mit gedrückter linker Maustaste nach oben zum Verkleinern oder nach unten zum Vergrößern.

Am linken Rand jedes Bereichs und über dem Formularentwurf ist jeweils ein *Lineal* zu sehen. Sie können das Lineal als Hilfe beim Zeichnen und Positionieren von *Steuerelementen* in einem Formular verwenden.

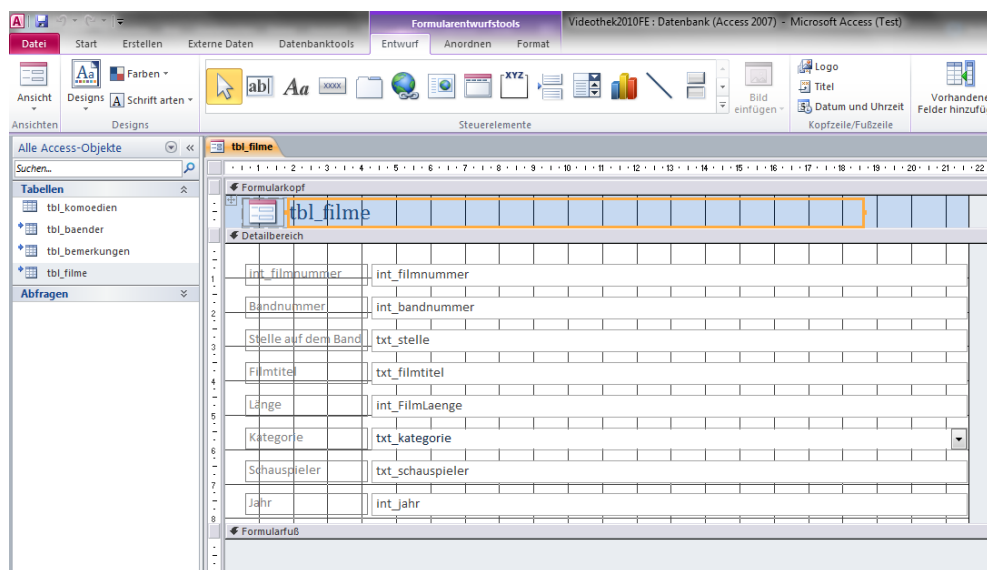
Übrigens: *Steuerelemente* heißt alles, was sich auf dem Formularentwurf tummelt; die Textfelder, die Felder mit den Daten, die Striche, Überschriften usw. Wenn Sie ein Steuerelement verschieben, erscheinen auf jedem Lineal Markierungen, die die Position des Steuerelements in Bezug auf die Unterteilungen des Lineals anzeigen.

### 5.3.3 Anpassen der Überschrift

Wie schon erwähnt wurde die Überschrift, die im Formulkopf steht, automatisch erstellt. Diese Überschrift überschreiben wir einfach mit dem Begriff, der uns richtig erscheint.

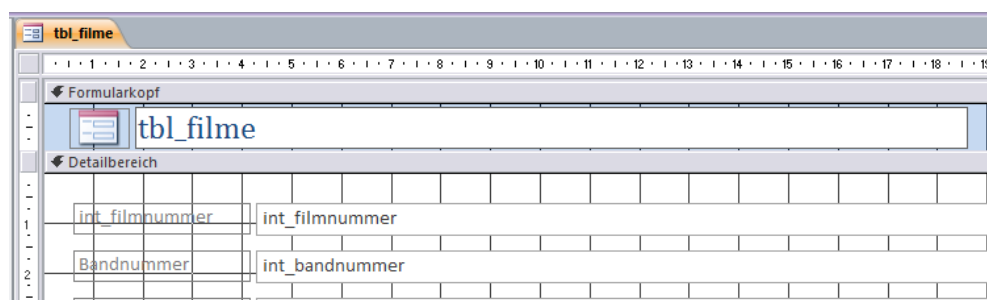
#### Übung:

1. Klicken Sie irgendwo in das Feld mit der Überschrift. Das Feld ist nun markiert bzw. ausgewählt.



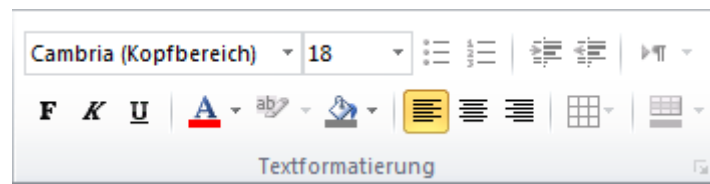
Abd. 5.8 markiertes Feld für die Überschrift

2. Klicken Sie noch mal innerhalb des markierten Bereichs um das Feld überschreiben zu können.



Abd. 5.9 Nun kann man selbst etwas reinschreiben

3. Um die Schrift zu formatieren, wie man das von Word her kennt, muss man das Feld wieder markieren (einmal außerhalb des Rahmens klicken und dann wieder in das Kästchen) und dann in der Multifunktionsleiste Schriftart, -größe oder –farbe einstellen.

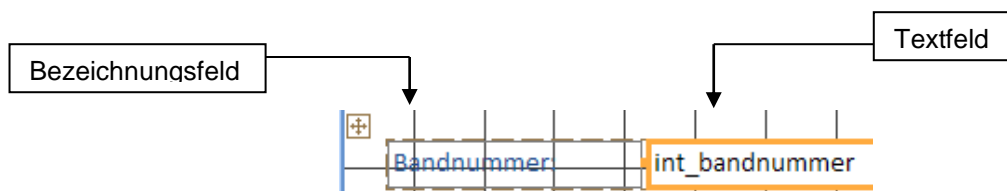


Abd. 5.10 Optionen zur Textformatierung

Genau so funktioniert es auch mit allen anderen Steuerelementen, die sich auf dem Formularentwurf befinden.

### 5.3.4 Markieren von Steuerelementen

Die übrigen Steuerelemente werden genauso markiert. Jedoch gibt es bei den meisten Feldern im Detailbereich des Formulars die Besonderheit, dass sie aus 2 Feldern bestehen, dem Bezeichnungsfeld und dem Textfeld.




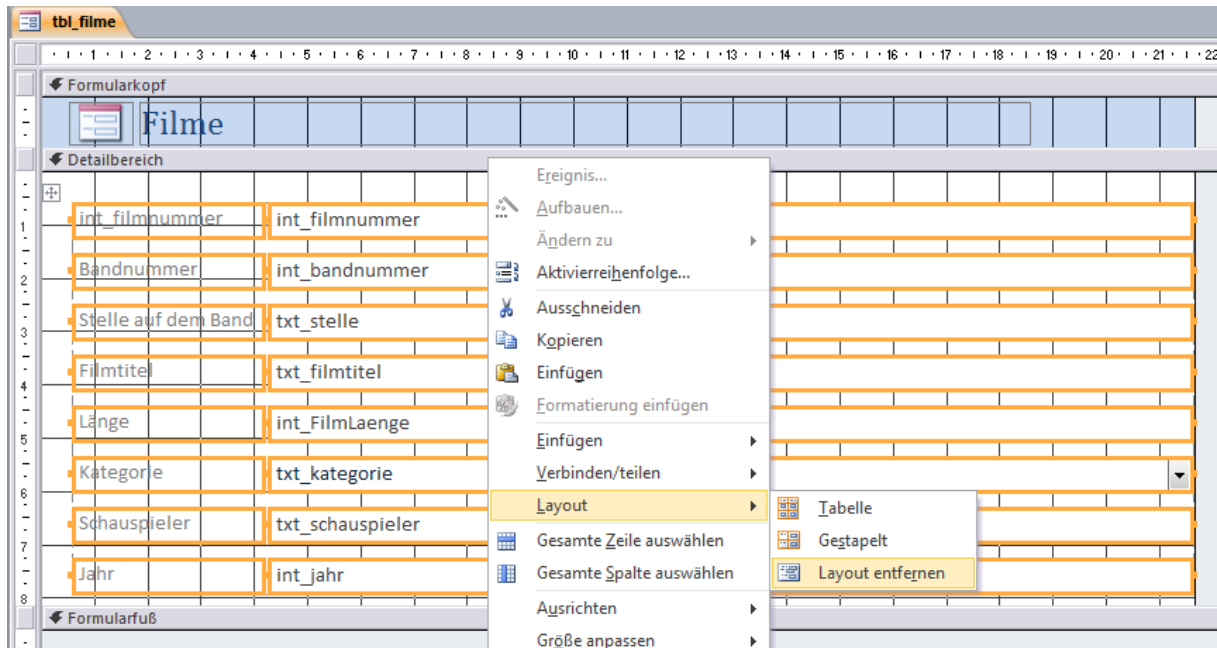
Abd.5.11 markiertes Steuerelement mit markiertem Textfeld (mit Layout)

Will man jedoch die Beschriftung des Bezeichnungsfeldes ändern, muss man erst mit einem weiteren Klick das Bezeichnungsfeld markieren.



Abd.5.12 markiertes Steuerelement mit markiertem Bezeichnungsfeld (mit Layout)

Eine weitere Besonderheit bei der Verwendung von Auto-Formularen seit Access 2007 sind die bereits mit den Formularen verknüpften Layouts. So können Sie bspw. gar nicht so ohne weiteres die Steuerelemente in unserem Formular an eine andere Position verschieben, das Layout wird Sie daran hindern. Um die Steuerelemente eines Auto-Formulars einzeln bearbeiten zu können, müssen Sie zuerst über das kleine Kreuz links oben neben den Steuerelementen  alle Felder markieren und danach über die rechte Maustaste den Befehl **Layout** → **Layout Entfernen** auswählen.



Abd.5.13 Layout entfernen

Nun sehen die Markierungen wieder etwas anders aus.




Abd.5.14 markiertes Steuerelement mit markiertem Textfeld (ohne Layout)

Die Ziehpunkte zur Größenänderung eines Steuerelements haben folgende Eigenschaften:

1. Mit der Maus können Sie die oberen und unteren Ränder eines Steuerelements zur Größenänderung nach oben und unten verändern.
2. Die Ränder an den rechten und linken Seiten sind für die Größenänderung nach rechts und nach links.
3. Die Ecken sind für beliebige Richtungs- und Größenänderungen.

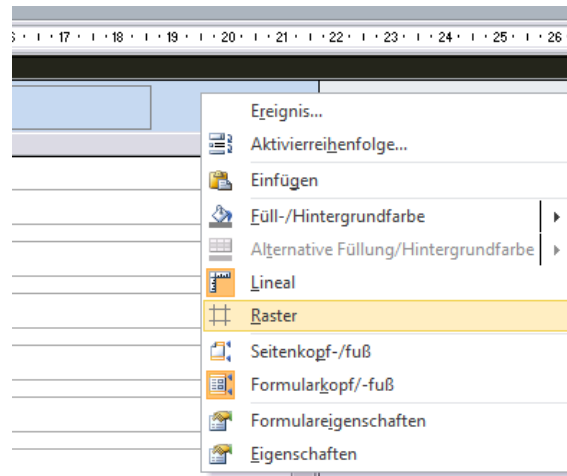
Mit den Ziehpunkten zum Verschieben lässt sich jedes Steuerelement unabhängig voneinander bewegen. Dazu muss man nur die dicken braunen Quadrate erwischen.

Es ist auch möglich, mehrere Steuerelemente gleichzeitig zu markieren. Klicken Sie dazu nacheinander alle Steuerelemente an, die markiert werden sollen, und halten dabei die UMSCHALT-Taste gedrückt. Während des Klickens sehen Sie ein Rechteck um die Steuerelemente.

Fährt man langsam und gezielt über die verschiedenen Bereiche des markierten Steuerelements wird man merken, dass sich je nach Bereich der Mauszeiger ändert. Ein Mauszeiger mit darunter liegendem Kreuz  verschiebt die einzelnen Elemente und mit den Pfeilen verändert man die Größe der Steuerelemente.

Übrigens kann man für mehr Übersichtlichkeit die Gitternetzlinien, die nur zum Anordnen der Steuerelemente hilfreich sind, über das Menüfenster, welches sich beim Klicken auf die rech-

te Maustaste öffnet, ausblenden. Dort den Befehl **Raster** wählen. Allerdings funktioniert das nur, wenn sich der Mauszeiger im Bereich des Formulkopfes befindet.




Abd.5.15 Menüfenster

Über dieses Menüfenster kann auch die Hintergrundfarbe über den Befehl **Füll-/Hintergrundfarbe** der drei Bereiche des Formulars geändert werden.


### 5.3.5 Verschieben von Steuerelementen

Sofern kein Layout festgelegt wurde können Steuerelemente beliebig verschoben und neu positioniert werden, wenn man sie markiert hat. Beim Markieren und Verschieben eines Textfeldes wird das ihm zugeordnete Bezeichnungsfeld automatisch mit verschoben.

Markieren Sie ein gewünschtes Textfeld oder Bezeichnungsfeld und stellen Sie den Mauszeiger dann auf den Rand der Markierung. Der Mauszeiger ändert seine Form zu dem Mauszeiger mit einem darunter liegenden Kreuz . Nun können Sie das Steuerelement beliebig dahin verschieben, wo sie es gern haben möchten.

Die Bezeichnungs- und Textfelder können jedoch auch unabhängig voneinander verschoben werden. Dieses Verfahren ist sehr einfach zu handhaben. Verwenden Sie dazu die "Ziehpunkte zum Verschieben", also die dicken Klötzchen in der linken oberen Ecke.

Folgende Arbeitsschritte sind notwendig:

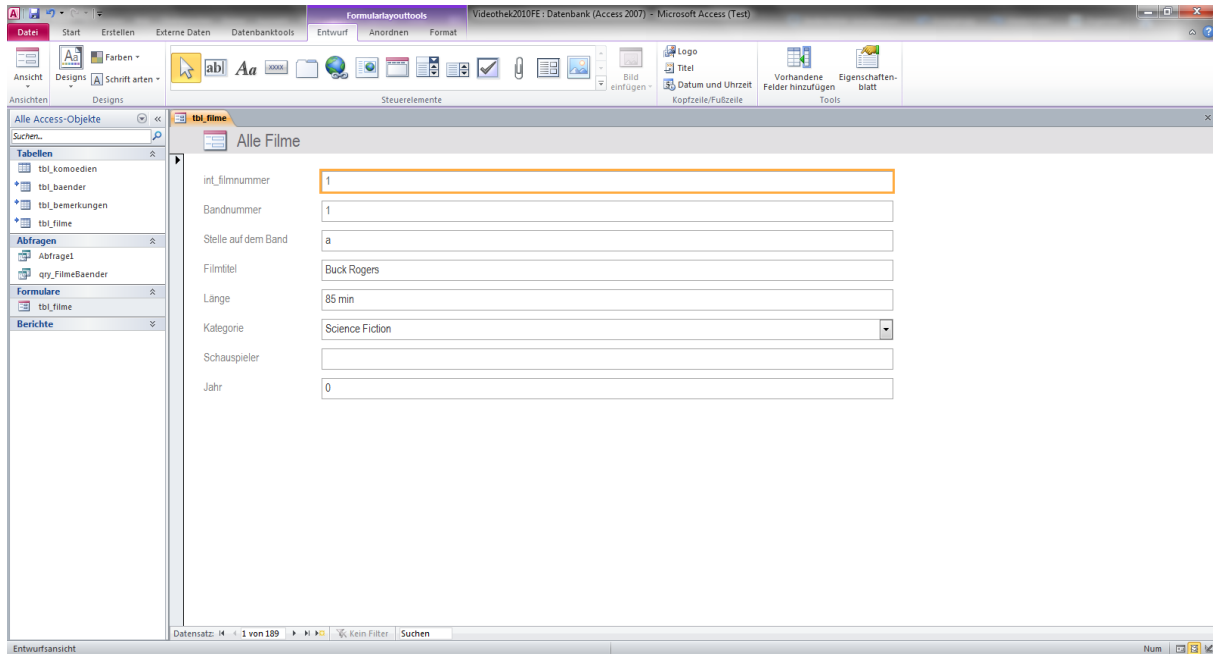
1. Markieren Sie das Steuerelement.
2. Positionieren Sie den Mauszeiger über dem Ziehpunkt zum Verschieben des Bezeichnungsfelds (oder Textfelds).  
Der Mauszeiger, der sich jetzt über dem Ziehpunkt befindet, ändert seine Form zu dem Mauszeiger mit darunter liegendem Kreuz .
4. Ziehen Sie den "Ziehpunkt zum Verschieben" und lassen die Maustaste wieder los, wenn das Bezeichnungsfeld an der gewünschten Stelle steht.

So, jetzt haben wir es endlich geschafft, uns einige einfache (und auch wichtige!) Arbeitsmethoden für das Erstellen eines Formulars anzueignen.



## Übung

1. Erstellen Sie auf der Basis von tbl\_filme ein neues Formular! Wählen Sie ruhig mal eine andere Formularform oder ein anderes Layout. Benutzen Sie die dabei erlernten Techniken zur Gestaltung des Layouts (Anordnung der Steuerelemente, Hintergrundfarbe anpassen, Schriftgröße einstellen etc..) und probieren Sie auch mal die Möglichkeiten der Layoutansicht aus.
2. Tragen Sie in das fertige Formular ca. 10 neue (von Ihnen ausgedachte) Datensätze ein.



**Abd. 5.16 mögliches Formular**

So, oder so ähnlich, könnte das Formular zum Schluss aussehen!

## 6 Berichte - Drucken in Access

Manchmal reicht es nicht, Daten, die man selektiert, geordnet und "sonst was" hat, nur auf dem Bildschirm nachzuvollziehen. Man braucht sie schwarz auf weiß. Für diese Aufgabe ist die Objektart **Berichte** vorgesehen. Zwar kann man auch aus den Formularen heraus drucken, aber dort fehlen ein paar Möglichkeiten, wie das Gruppieren und Sortieren, das hier geboten ist.

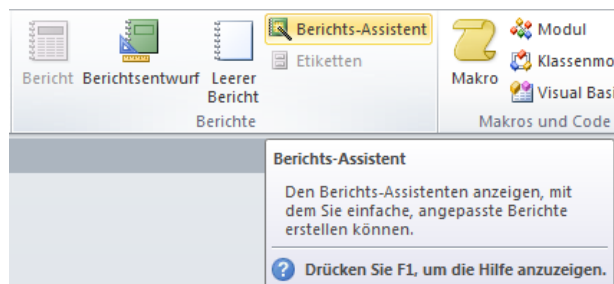
Das Drucken in *Access* kann sich beliebig schwierig gestalten. Da aber das Prinzip den Formularen ähnelt, werden in diesem Skript nur zwei Beispiele durchgenommen, um anzudeuten was alles geht.

### 6.1 Listendruck mit dem Berichtsassistenten

#### 6.1.1 Der Assistent, was leistet er

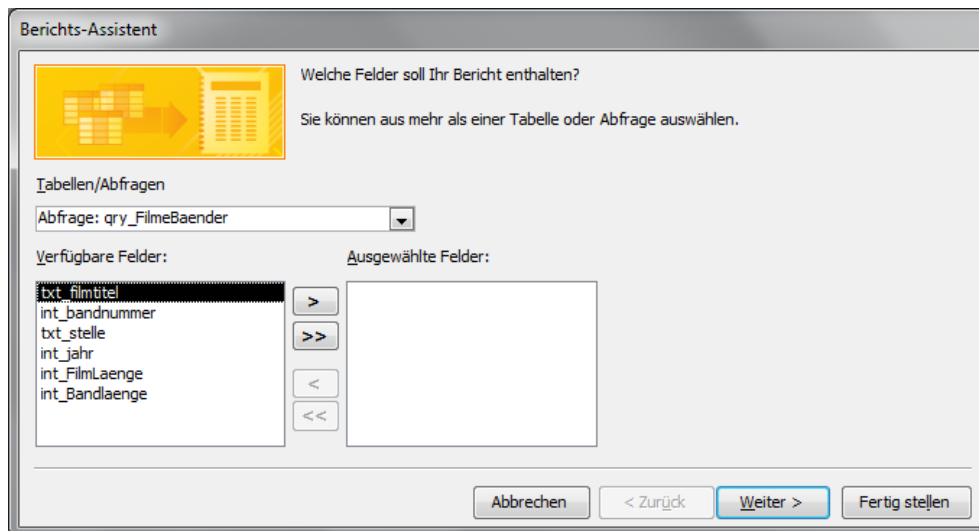
Für das Drucken braucht man Daten, logisch. Also ist häufig der erste Schritt das Erstellen einer Abfrage, in der die Daten, die man nachher drucken möchte, stehen. In unserem Fall wollen wir nur eine Kurzliste der Filme, der Filmbänder, der genauen Stelle, des Erscheinungsjahrs und Filmlänge. Diese Abfrage gibt es schon: **qry\_FilmeBaender**. Falls doch nicht, muss an dieser Stelle eine solche Abfrage, die alle Felder enthält und nur die Bänder, auf denen etwas drauf ist, hergestellt werden.

Diese Abfrage dient uns als Grundlage für den Bericht, daher markieren wir die Abfrage mit einem einfachen Mausklick im Navigationsbereich. Als nächstes wählen Sie in der Multifunktionsleiste auf der Registerkarte **Start** den Befehl **Berichts-Assistent**.



Abd. 6.1 Symbol für den Berichts-Assistenten in der Multifunktionsleiste

Wir wählen also den Berichts-Assistenten nachdem wir davor die Abfrage **qry\_FilmeBaender** markiert haben. Es öffnet sich das folgende Fenster:




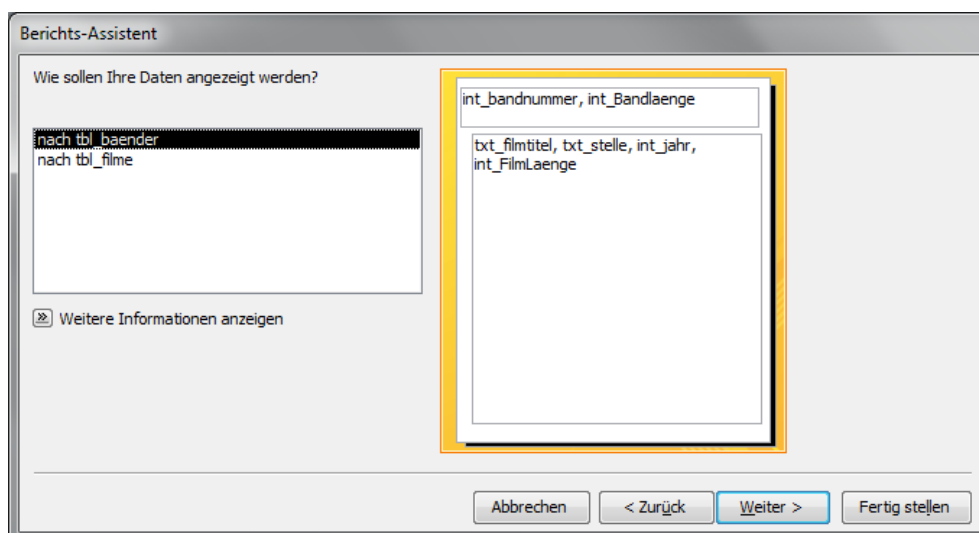
Abd. 6.2 Berichts-Assistent 1. Schritt

Hier kann man die Felder, die man im Bericht haben möchte auswählen, was man auch dem Assistenten entnehmen kann.

Will man alle Felder im Bericht übernehmen, kann man auf den Doppelpfeil (>>) drücken, ansonsten markiert man das gewünschte Feld und klickt auf den einfachen Pfeil (>). Will man Felder wieder zurücknehmen, funktioniert das genauso mit den Pfeilen, die in die entgegengesetzte Richtung zeigen.

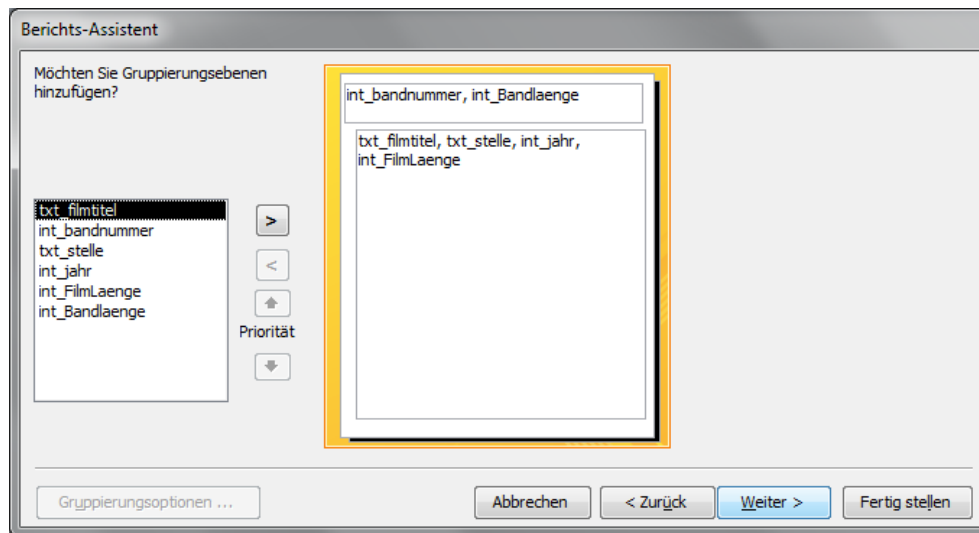
Falls man mehrere Felder aus mehreren Tabellen und Abfragen in den Bericht einschließen möchte, klickt man nicht auf **Weiter** oder **Fertig stellen**, nachdem man die Felder aus der ersten Tabelle oder Abfrage im Berichts-Assistenten ausgewählt hat. Man wählt dann einfach im gleichen Fenster die nächste Abfrage oder Tabelle aus dem Auswahlmenü aus und kann wiederum Felder hinzufügen oder löschen. Dies wiederholt man solange, bis der Bericht alle benötigten Felder enthält.

Unser Bericht soll erst einmal aus allen aufgeführten Feldern der qry\_FilmeBaender bestehen, außer dem Feld int\_FilmLaenge. Danach geht es weiter mit .



Abd. 6.3 Berichts-Assistent: 2. Schritt

Das Programm bietet einem nun verschiedene Optionen zur Gruppierung der Daten. Hier kann man so genannte Gruppierungsebenen bestimmen. Um die Bestimmung dieser Gruppierungsebenen geht es auch im nächsten Schritt des Assistenten, dort haben Sie lediglich mehr Auswahlmöglichkeiten und Sie können mehr als nur eine Gruppierungsebene anlegen (maximal 4 Gruppierungsebenen pro Bericht).




Abd. 6.4 Berichts-Assistent: 3. Schritt

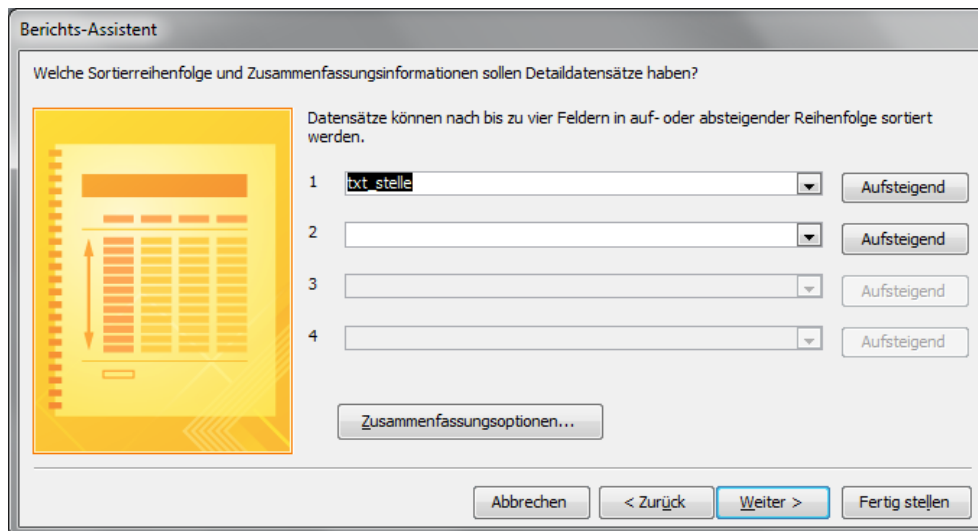
Eine Gruppierung nach Bandnummer bewirkt zum Beispiel, dass nicht vor allen Filmen die Bandnummer steht, sondern nur einmal auftaucht und die Zugehörigkeit der Filme zu den Bändern durch ihre Anordnung deutlich wird. Dies ist in Abd. 6.5 (Seitenansicht) dargestellt.

1	Krieg der Eispiraten	o	180
	Buck Rogers	o	180
2	Scotland Yard jagt Dr. Mabuse	o	180
	Auf der Jagd nach dem Juwel vom N	o Michael Douglas	180
	Stirb langsam II	o Bruce Willis	180

Abd. 6.5 Gruppierung nach Bänder

Mit den Prioritäts-Tasten (< und >) kann man also bestimmen, welche Felder Priorität haben sollen und welche nicht. Anders ausgedrückt kann man die Feldnamen nach der gewünschten Priorität sortieren. Wir klicken mal wieder auf .

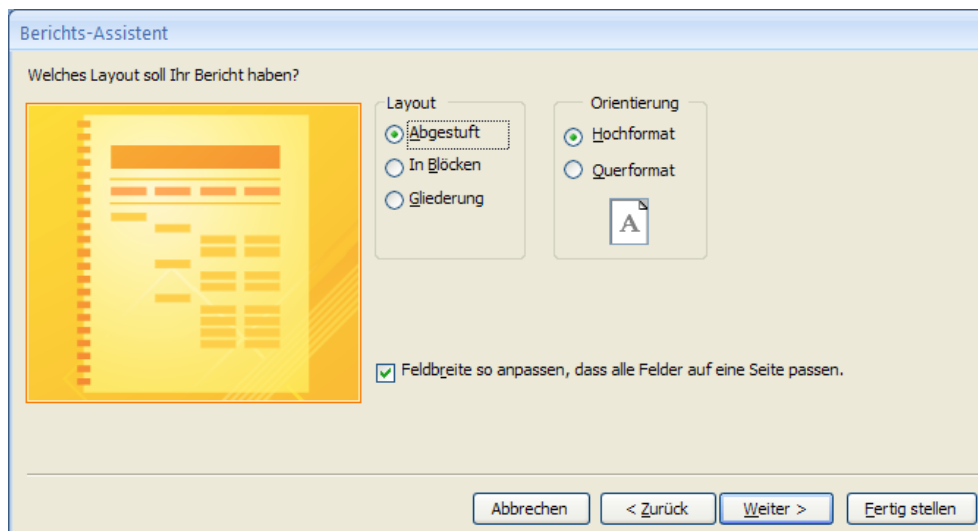
Im folgenden Fenster können wir bestimmen, nach welchen Kriterien die Datensätze sortiert werden sollen. Durch die Gruppierung werden unsere Filme bereits automatisch nach ihrer Bandnummer sortiert. Eine Sortierung nach dem Feld **txt\_stelle** wäre noch sinnvoll.



Abd. 6.6 Berichts-Assistent: 4. Schritt

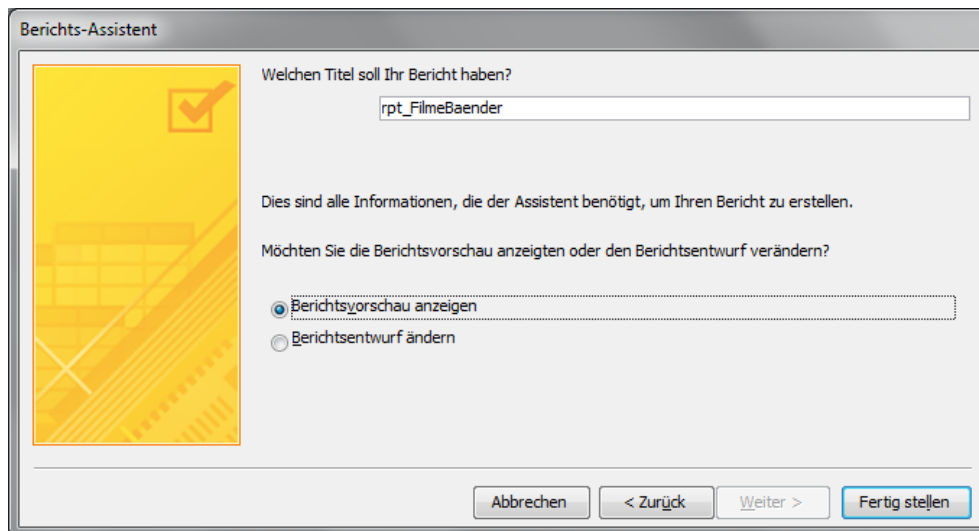
Wie Sie in Abd. 6.6 sehen, können bis zu 4 Sortierkriterien angegeben werden. Sollte dies nicht ausreichen, können über die Layout- und die Entwurfsansicht später noch weitere Sortiermerkmale eingestellt werden.

Der nächste Schritt (Abd. 6.7) des Assistenten erlaubt ihnen ein Layout für ihren Bericht zu wählen. Wir wählen für unseren Bericht das Layout *Abgestuft* mit der Orientierung *Hochformat*.



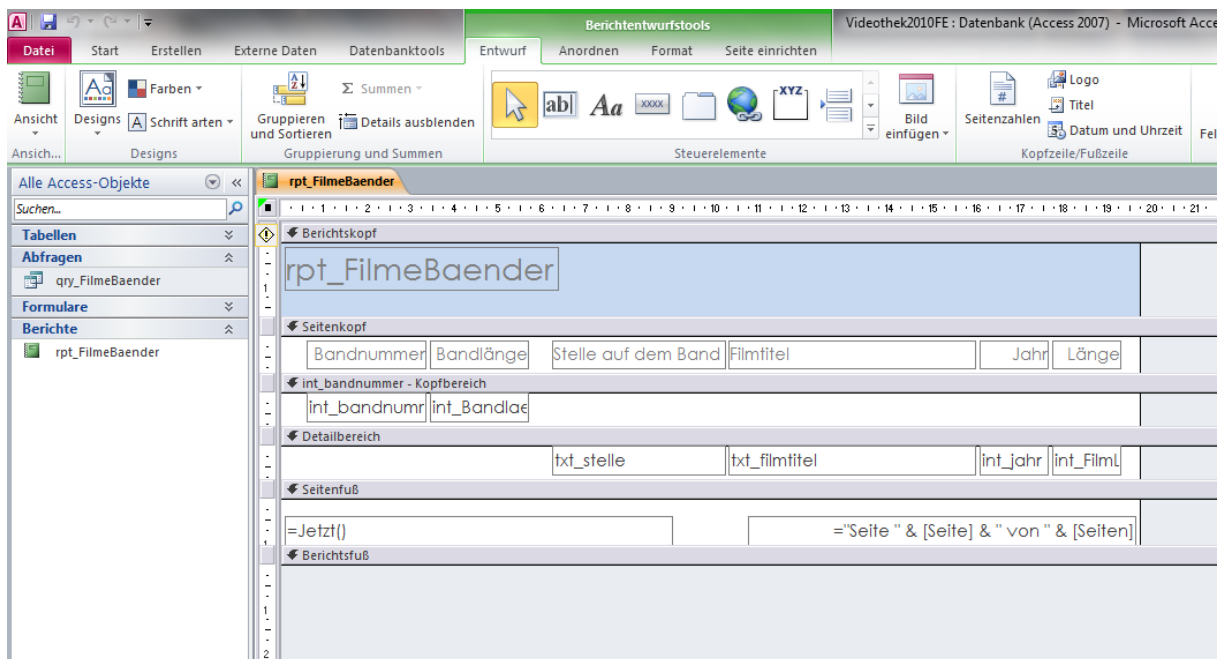
Abd. 6.7 Berichts-Assistent: 5. Schritt

Der letzte Schritt erfordert, dass man dem Bericht einen Namen gibt. Wir nennen unseren Bericht *rpt\_FilmeBaender*.



Abd. 6.8 Berichts-Assistent: 6. Schritt

Weiterhin kann man wählen, ob man ein Bericht in der Vorschau sehen will, oder gleich in die Entwurfsansicht wechselt, in der man noch selber Änderungen vornehmen kann. Wählt man das letztere, braucht Access etwas um den Bericht zu erstellen. Das Ergebnis sieht dann so aus:



Abd.6.9 Bericht in der Entwurfsansicht

Man erkennt sofort die Ähnlichkeit mit der Entwurfsansicht für Formulare. Das Prinzip ist bei Berichten und Formularen identisch, beide Objektarten dienen der Darstellung von Daten, jedoch unterscheiden sich die Ausgabemedien. Formulare dienen der Darstellung am Bildschirm, während Berichte i.d.R. für den Druck von Daten vorgesehen sind.

Wenn Sie die Ansicht in die Seitenansicht ändern, erhalten Sie eine Druckvorschau des Berichts.

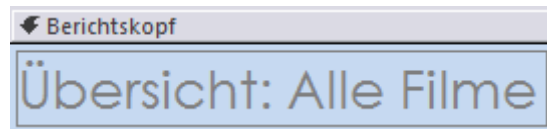
## 6.1.2 Text ändern und einfügen - Berichtskopf und Seitenkopf

Keht man in die Entwurfsansicht zurück, ist im Berichtskopf die Überschrift wieder zu finden, was so aussieht:



Abd.6.10 Berichtskopf

Die Überschrift soll jetzt in *Übersicht: Alle Filme* umgeändert werden. Wenn man das Kästchen markiert und einmal in die Multifunktionsleiste schaut, sieht man, dass sich dort lauter Zeichenformatierungssymbole anbieten. So kann der Text auch noch kursiv, unterstrichen und 20 Punkt groß eingestellt werden, wenn das nicht schon gemacht worden ist. Passt die Überschrift aufgrund ihrer Vergrößerung nicht mehr in das Kästchen, kann man es mit der Maus an den Rändern packen und größer ziehen. Eigentlich alles wie bei den Formularen (Kapitel 5).



Abd. 6.11 Der neue Berichtskopf

Dieser ganze Bereich, der so genannte *Berichtskopf*, erscheint **einmal** am Anfang des Drucks und hat damit seine Schuldigkeit getan.

Danach schließt sich der *Seitenkopf* an. Dieser nimmt Text oder auch Felder auf, die am Anfang jeder Seite erscheinen sollen. Bei unserem Beispiel sind dort die Spaltenbezeichnungen eingetragen.



Abd. 6.12 Seitenkopf und Detailbereich

Es folgt ein besonderer Bereich in Berichten, ein sogenannter Kopfbereich. Dieser Kopfbereich ermöglicht die Gruppierung der Daten nach der Bandnummer und heißt deshalb auch so. In dem Bereich werden sämtliche Felder aus der Tabelle *tbl\_baender* angezeigt, also die Bandnummer und die Bandlänge.


Der Kopfbereich bewirkt, dass für jedes Band die Bandnummer nur einmal angezeigt wird und darunter die Filme, die sich auf diesem Band befinden, erscheinen.





Abd. 6.13 Seitenkopf und Detailbereich

Unter dem Kopfbereich folgt der Detailbereich, den man wieder bereits von den Formularen kennt. Wie man hier seinen Namen oder sonstigen Text einfügen kann, soll in der folgenden Übung erläutert werden (wobei dies schon mal im Kapitel Formulare durchgenommen worden ist, also eigentlich nichts Neues ist):

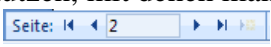
## Übung:

Ziehen Sie mit der Maus den Balken *Detailbereich* weiter nach unten, so dass im Seitenkopf mehr Platz entsteht. Dazu muss der Mauszeiger so aussehen: 

Klickt man in der Multifunktionsleiste auf der Registerkarte **Entwurf** auf  kann man im Entwurfsbereich ein Kästchen beliebiger Größe erstellen, in das man beliebigen Text (z.B. ihren Namen) schreiben kann. Klickt man mit der Maus außerhalb des Kästchens, kann man den Text beliebig formatieren.

Also merke: Das  ist zum Hinzufügen von beliebigem Text in den Bericht gedacht.

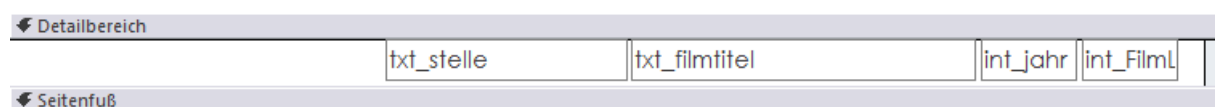
Schreiben Sie in ein solches Feld, das Sie irgendwo im Seitenkopf platzieren, Ihren Namen.

Beim Anschauen in der Seitenansicht sollten Sie auch wirklich durch die Seiten blättern, indem Sie unten in dem Fenster die Pfeile benutzen, mit denen man sonst von Datensatz zu Datensatz blättern kann. Jetzt gelten die Pfeile  nämlich für das Umblättern pro Seite, bzw. dazu, um an die erste oder letzte Stelle des Berichts zu springen.

Wenn alles geklappt hat, dann steht auf jedem Blatt ganz oben Ihr Name.

### 6.1.2.2 Felder verschieben und einfügen - Detailbereich

Der nächste Bereich des Berichts heißt *Detailbereich* (Abd.6.14) und ist zuständig für die eigentliche Präsentation der Daten. Hier werden die Felder angeordnet, mit Eigenschaften versehen und aufbereitet.



Abd.6.14 Detailbereich der Entwurfsansicht

Eine kleine Bemerkung am Rande. Man kann mit Hilfe der Maus die verschiedenen Bereiche beliebig groß ziehen. Auch nach rechts hat man die Möglichkeit, das Blatt größer zu machen. Deshalb ist oberhalb und an der linken Seite des Arbeitsbereichs ein Lineal angebracht, das einem für die Ausdehnung Anhaltspunkte gibt.

Der darunter folgende *Seitenfuß* ist das Gegenstück zum *Seitenkopf*, also vergleichbar mit einer Fußzeile.

Man kann nun nicht nur die Texte beliebig verändern und hinzufügen, wie das bei dem vorherigen Punkt gemacht wurde, sondern das gleiche klappt mit den Spaltennamen auch. Klickt man auf **Bandnummer** (egal an welche Stelle) sieht das Ergebnis in unserem Beispiel so aus:



Abd. 6.15 Markiertes Bezeichnungs- und Textfeld

Der untere Teil des Gespanns beinhaltet das eigentliche Feld, also Daten wie die unterschiedlichen Bandnummern, Filmtitel oder Filmstellen.



Die Markierung zeigt an, dass die beiden Felder verbunden sind, d. h., wenn eines gegriffen wird verschiebt man mit gedrückter linker Maustaste das andere Feld gleich mit. Falls wirklich mal nur die Überschrift oder nur der Feldinhalt an einen anderen Platz verschoben werden soll, dann muss man die **dicken braunen Klötze** "anpacken", wobei sich der Mauszeiger zu der Hand mit ausgestrecktem Zeigefinger verändern muss. Dann werden die beiden Komponenten voneinander gelöst.

Um ein neues Feld einzufügen, kann man wieder auf die **Toolbox** zurückgreifen. Allerdings ist das Hinzufügen von neuen Feldern im Nachhinein nur eine Notlösung. Richtig ist es, dieses Feld schon bei der Erstellung des Berichts durch den Assistenten hinzuzufügen. Es kann aber trotzdem sein, dass man mal etwas vergisst. Also:

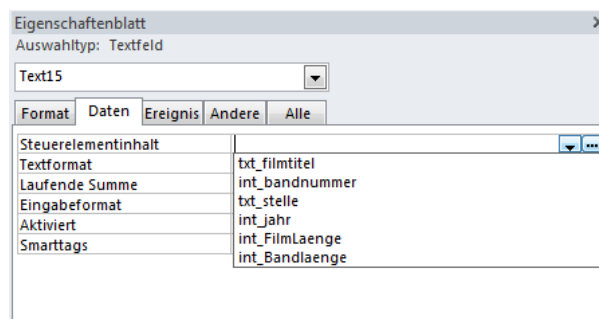
In der Toolbox gibt es einen Knopf dafür . Merke: Dieser Button ist für das Hinzufügen von Feldern (im Gegensatz zu Text!). Wenn man den Knopf drückt und die Maus auf das Berichtsblatt schiebt und dort einmal klickt, erhält man folgendes Muster:



Abd. 6.16 neu eingefügtes Textfeld- noch ungebunden

Es kann sein, dass nicht immer die Zahl 15, sondern irgendeine andere Zahl dort steht, aber das ist egal. Der rechte Teil des neuen Feldes ist dagegen noch weiter zu bearbeiten. Das Programm weiß nämlich gar nicht, **welches Feld hier ausgedruckt werden soll!** Na klar, woher auch.

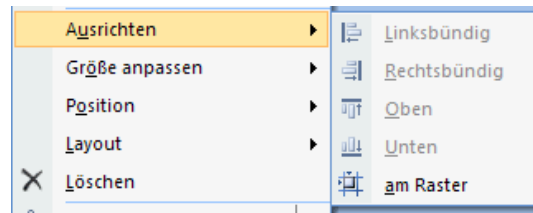
Deshalb muss man mit einem Rechtsklick über den Befehl **Eigenschaften** das Eigenschaftfenster öffnen, welches am rechten Bildschirmrand erscheint.



Abd.6.17 Steuerelementinhalt festlegen

Ein bisschen blöd ist, dass das Ding Textfeld heißt, das führt leicht zu Verwechslungen, aber jetzt weiß man ja was gemeint ist. Von all diesen vielen Eigenschaften interessiert uns im Moment eigentlich nur eine, nämlich **Steuerelementinhalt**. Klickt man dort hinein, hat man rechts wieder den Pfeil nach unten, der auf eine Liste hinweist, die man öffnen kann. Und richtig: macht man es, dann erscheinen alle Feldnamen aus der zugrunde liegenden Abfrage! In dem Moment, in dem man sich ein Feld auswählt, übernimmt **Access** es in den Bericht.

Damit man die Felder ohne Layout halbwegs bündig auf der Entwurfsfläche verschieben kann, sollte man sich das Raster einrichten. Dies geht über das Fenster, welches sich beim Klicken auf die rechte Maustaste im Entwurfsbereich öffnet. Dort einfach auf Raster drücken. Um die Felder am Raster auszurichten, markieren Sie die auszurichtenden Steuerelemente und wählen nach einem Rechtsklick den Befehl **Ausrichten → am Raster**.



Abd.6.18 Steuerelemente am Raster ausrichten

Um nur bestimmte Felder auszurichten, markiert man zuerst das Feld, von dem aus ein anderes Feld ausgerichtet werden soll. Mit gedrückter SHIFT - Taste markiert man nun das auszurichtende Feld und gibt dann den Befehl *Ausrichten* im Menü. Es erscheint das bereits bekannte Untermenü aus dem man Links-, Rechtsbündig, Oben oder Unten oder am Raster wählen kann. Damit ist das genaue Verschieben der Felder ganz einfach.

Um noch genauer zu arbeiten, können Sie die Position der einzelnen Steuerelemente auch im Eigenschaftfenster festlegen. Den Befehl Eigenschaften erreichen Sie auch wieder über einen rechten Mausklick auf das Steuerelement, das Sie bearbeiten wollen.

Übrigens kann man alle Objekte (Linien oder ähnliches) ganz einfach entfernen, indem Sie sie markieren und dann auf ENTF drücken.

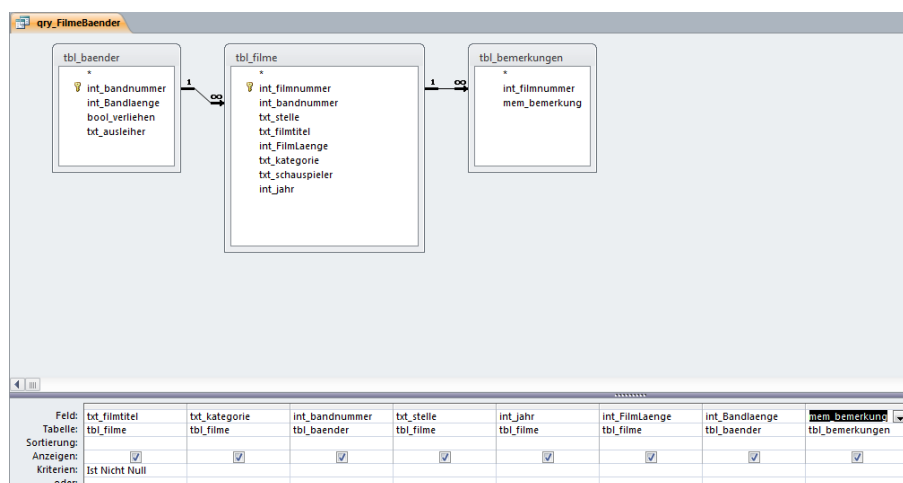
Der unter dem *Detailbereich* folgende *Seitenfuß* ist das Gegenstück zum *Seitenkopf*, also vergleichbar mit einer Fußzeile.

Im Bereich *Seitenfuß* stehen Daten, die am Ende jeder Seite auftauchen sollen. In unserem Beispiel stehen u.a. die Seitenzahlen dort, was man an dem "SEITE" & [SEITE] usw. erkennen kann.. Man kann die Seitenzahlen aber auch in den Seitenkopf einfügen. Je nachdem was man für sinnvoll hält. In der Entwurfsansicht verschiebt man dazu einfach die Kästchen, wie alle anderen auch (siehe Kapitel 5.0 Formulare).

### Übung

Damit wir eine sinnvolle Übung hinbekommen, müssen wir eine kleine Wiederholung einlegen.

1. Fügen Sie der Abfrage *qry\_FilmeBaender* das Feld **Kategorie** hinzu. Positionieren Sie das Feld Kategorie hinter Filmtitel.
2. Fügen Sie der gleichen Abfrage das Feld **Bemerkung** aus der Tabelle *tbl\_bemerkungen* hinzu.



Abd.6.19 Abfrage qry\_FilmeBaender

3. Damit nur Bänder angezeigt werden, auf denen auch was drauf ist, sollten Sie in der Abfrage auch noch ein Kriterium festlegen. Zu diesem Zweck bietet es sich an, nur Bänder anzeigen zu lassen, die auch eine Bandnummer besitzen.

Feld:	txt_filmtitel
Tabelle:	tbl_filme
Sortierung:	
Anzeigen:	<input checked="" type="checkbox"/>
Kriterien:	Ist Nicht Null
oder:	

Abd.6.20 Kriterium zur Anzeige der Bänder

4. Speichern und schließen Sie die Abfrage.

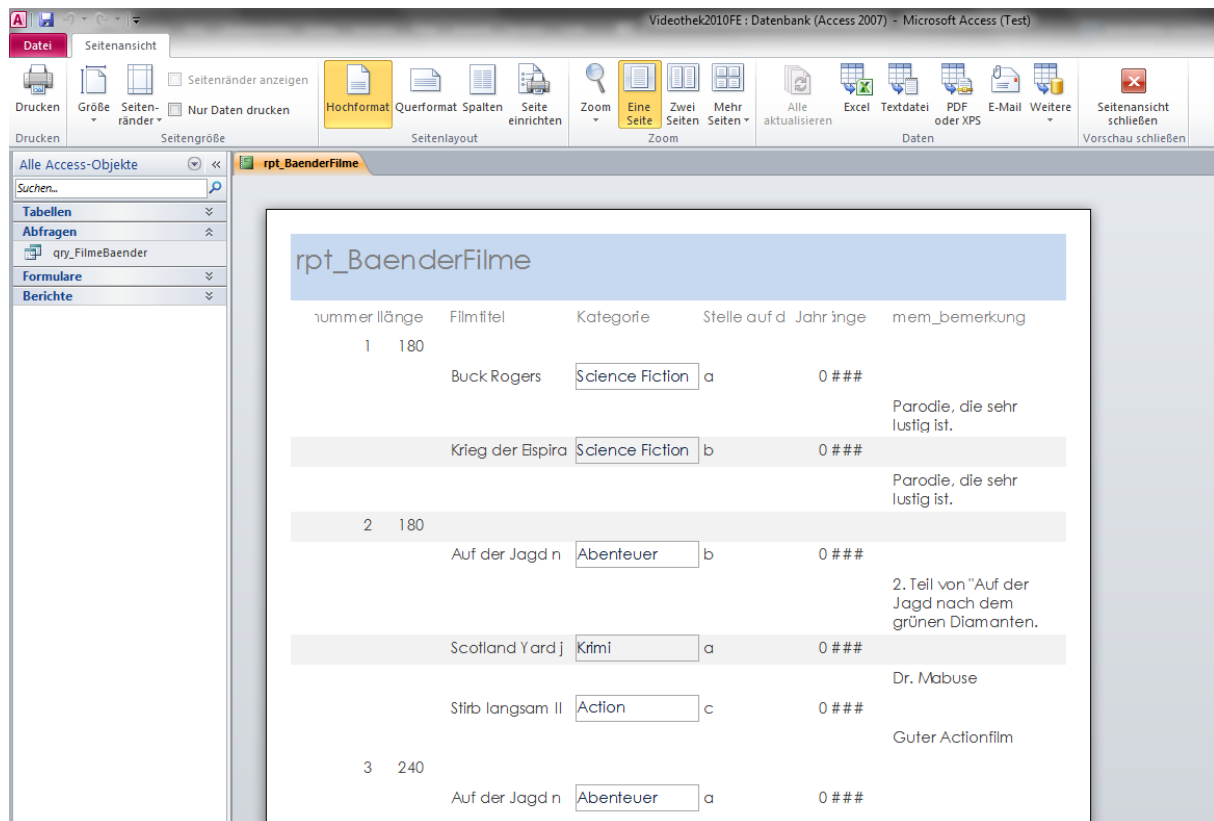
Klappt das noch alles? Sonst gucken Sie noch einmal nach. Wenn Sie damit fertig sind, löschen Sie bitte erst einmal alle Berichte, die im Navigationsbereich erscheinen.

Nun erstellen Sie mit Hilfe des Assistenten, wie oben beschrieben, einen neuen Bericht, der auf der eben angepassten Abfrage *qry\_FilmeBaender* basiert. Wichtig ist, dass Sie zwei Gruppierungsebenen hinzufügen, eine für Bandnummer und die andere für Filmtitel.

Abd.6.21 Gruppierungsebenen einstellen

Nennen Sie den Bericht im letzten Schritt des Assistenten wieder *rpt\_FilmeBaender*.

Gemein, alles noch mal zu machen, was? Aber Übung macht den Fortgeschrittenen! So könnte Ihr Bericht in der Seitenansicht aussehen:



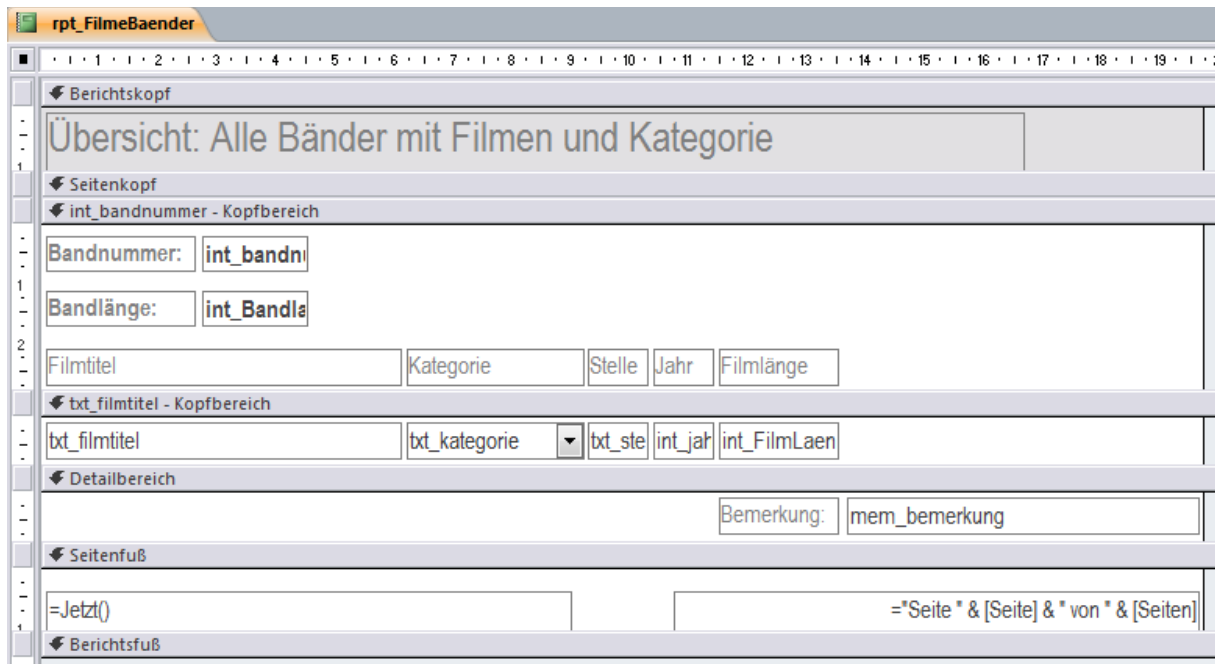
Abd.6.22 Bericht aus der Übung in der Seitenansicht

Da gibt es noch einiges zu tun, bevor wir diesen Bericht drucken können. Unter anderem werden dort statt der **Filmlänge** nur Raute-Zeichen (#) angezeigt. Das passiert immer dann, wenn der Platz für die dort vorgesehene Zahl nicht ausreicht. Führen Sie also folgende Übung durch:

### Übung

1. Verschieben Sie die Beschriftungsfelder für Bandnummer und Bandlänge in den Kopfbereich von Bandnummer. Dadurch können alle anderen Felder weiter links angeordnet werden, um den Platz der Seite besser auszunutzen.
2. Verschieben Sie die Beschriftungsfelder für Filmtitel, Kategorie, Stelle, Jahr und Filmlänge ebenfalls in den Kopfbereich von Bandnummer. Dadurch werden diese Beschriftungsfelder bei jedem Band angezeigt.
3. Das Beschriftungsfeld für Bemerkung können Sie ebenfalls im Kopfbereich von Bandnummer positionieren. Oder Sie verschieben dieses Bezeichnungsfeld in den Detailbereich.
4. Ändern Sie die Überschrift des Berichts in „Übersicht: Alle Bänder und Filme“.
5. Richten Sie die Felder im Bericht links- und rechtsbündig aus. Passen Sie auch die Größe der Felder an, damit alle Inhalte richtig angezeigt werden und um ein möglichst homogenes Aussehen zu erreichen. Verwenden Sie dazu die Optionen die erscheinen, wenn man ein oder mehrere Felder markiert und dann mit der rechten Maustaste das Kontextmenü öffnet. Vor allem die Optionen **Layout**, **Ausrichten**, **Größe anpassen** und **Position** sind in diesem Zusammenhang hilfreich.
6. Entfernen Sie die Rahmen, die bei einigen Feldern sichtbar sind.

Die folgenden Abbildungen zeigen das mögliche Resultat Ihrer Arbeit:



Abd.6.23 So könnte der Bericht in der Entwurfsansicht aussehen ...

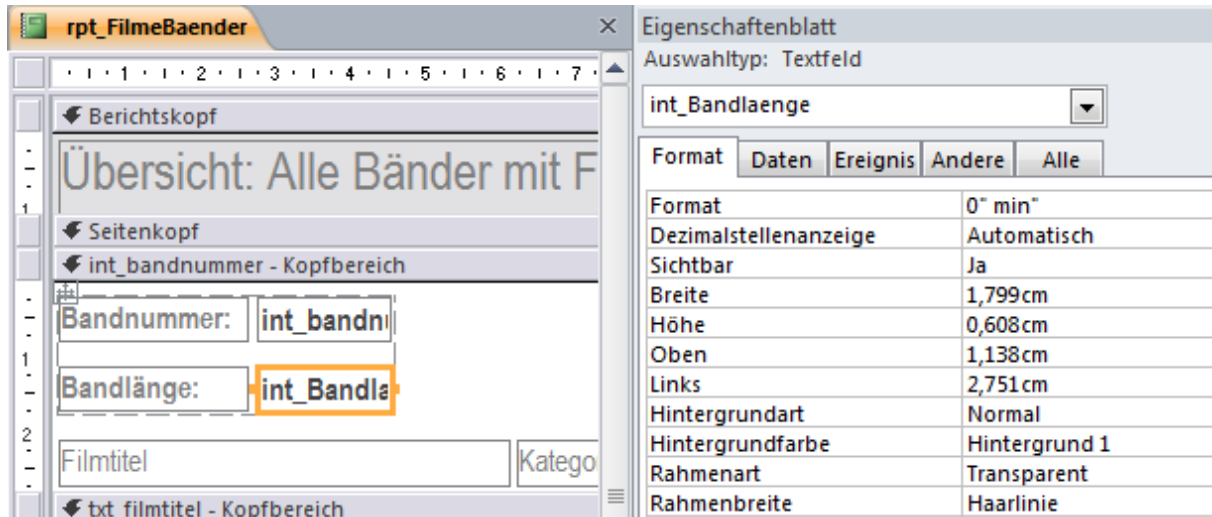


Abd.6.24 ... und so in der Seitenansicht

Nun verteilen sich die Felder gut über die gesamte Breite des Blattes. Die Zahlen in den Spalten Filmlänge und Bandlänge werden nun auch richtig angezeigt.

Die Formatierung der Werte bei Bandlänge sollte jetzt noch angepasst werden, damit dort ebenfalls wie bei der Filmlänge die Einheit hinter der Zahl (min) angezeigt wird. Das geht ganz einfach:

7. Markieren Sie das Feld **int\_Bandlaenge** in der Entwurfsansicht oder in der Layoutansicht und öffnen Sie das Eigenschaftenfenster. Am einfachsten klappt das mit einem Doppelklick auf das Feld. Bei den Eigenschaften tragen Sie in der Registerkarte **Format** bei der Eigenschaft **Format** das Folgende ein:  
**0“ min“**



Abd.6.25 Formateinstellung des Feldes Bandlänge

Zum Ende wollen wir noch das kleine Problem in den Griff bekommen, das sich bei unseren Restzeitberechnungen in Abfragen ergeben hat. Noch einmal: Es geht auch im Abfragebereich, aber über die Berichterstellung ist es einfacher. Das Problem bestand darin, dass **Access** auch bei Bändern, bei denen zwei Filme Platz beanspruchten, nicht bereit war, diese erst zu addieren und die Summe dann von der gesamten Bandlänge abzuziehen. Um das zu erreichen, müssen die Daten bereits entsprechend *Gruppirt* sein. Unsere Gruppierung nach der Bandnummer eignet sich perfekt dazu. Also weiter mit der Übung:

8. Unser Bericht besitzt in der Entwurfsansicht einen besonderen Bereich, nämlich **int\_bandnummer - Kopfbereich**. Das ist die obere Abgrenzung für die Gruppierungen. Darunter befindet sich ein weiterer Kopfbereich von einer weiteren Gruppierungsebene nach Filmtitel und danach folgt der *Detailbereich* mit seinen Daten. Allerdings benötigen wir jetzt auch noch einen Fußbereich **int\_bandnummer - Fußbereich**. Den erhalten Sie, indem Sie in der Multifunktionsleiste die Schaltfläche **Gruppieren und sortieren** betätigen.



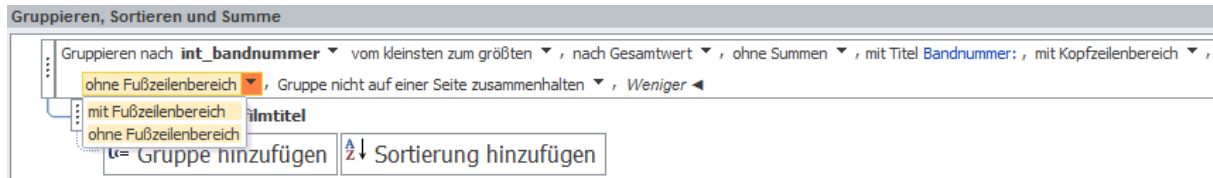
Abd.6.26 Schaltfläche Gruppieren und sortieren in der Multifunktionsleiste

9. Im unteren Bereich des Bildschirms öffnet sich ein besonderer Bereich namens **Gruppieren, Sortieren und Summe**. In diesem Bereich müsste bereits unser Feld **int\_bandnummer** als Gruppierungsebene auftauchen. Klicken Sie in der Zeile, in der **int\_bandnummer** steht, auf den Button **Mehr ▶**.



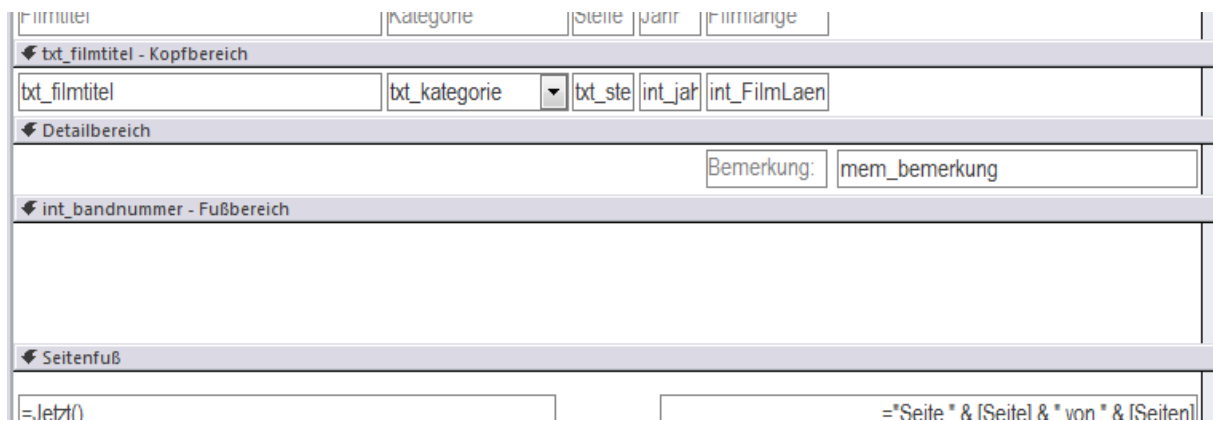
**Abd.6.27 Gruppieren, Sortieren und Summe**

10. Daraufhin können Sie statt des eingestellten „ohne Fußzeilenbereich“ einfach „mit Fußzeilenbereich“ auswählen.



**Abd. 6.28 Fußbereich der Gruppierungsebene sichtbar machen**

In der Entwurfsansicht sollte nun der neue Fußbereich **int\_bandnummer - Fußbereich** unter dem Detailbereich angezeigt werden:



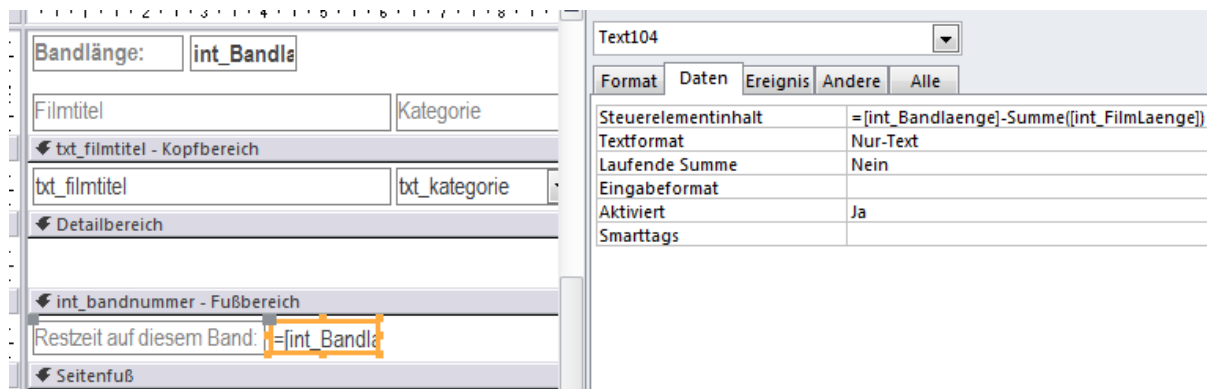
**Abd. 6.29 Der Fußbereich ist nun in der Entwurfsansicht sichtbar**

Der eigentliche Sinn dieser ganzen Aktion war, dass wir uns Platz für ein neues Feld schaffen, in dem **Access** die verbleibende freie Bandzeit ausrechnet und auch im Bericht anzeigt. Dafür benötigt man eigentlich nur noch eine Formel. Was sonst. Die Formel soll bewirken, dass die Filmlängen pro Band zusammengezählt werden.

11. Wir benötigen ein neues Feld mit dem Feldnamen "Restzeit" in dem Fußbereich der Gruppierung, also in **int\_bandnummer - Fußbereich**. Fügen Sie dort über das Symbol **ab|** in der Multifunktionsleiste ein neues Textfeld ein. In das Bezeichnungsfeld schreiben Sie etwas passendes, wie bpw. „Restzeit auf diesem Band:“.

12. Nun schreiben wir in das neu geschaffene Feld eine Formel, die die Rechnung Bandlänge Minus Summe der Filmlänge aller Filme auf diesem Band = Restzeit beinhaltet. Genau das wollen wir. Die Formel dafür sieht folgendermaßen aus:  
**=([int\_Bandlaenge]-Summe([int\_FilmLaenge]))**

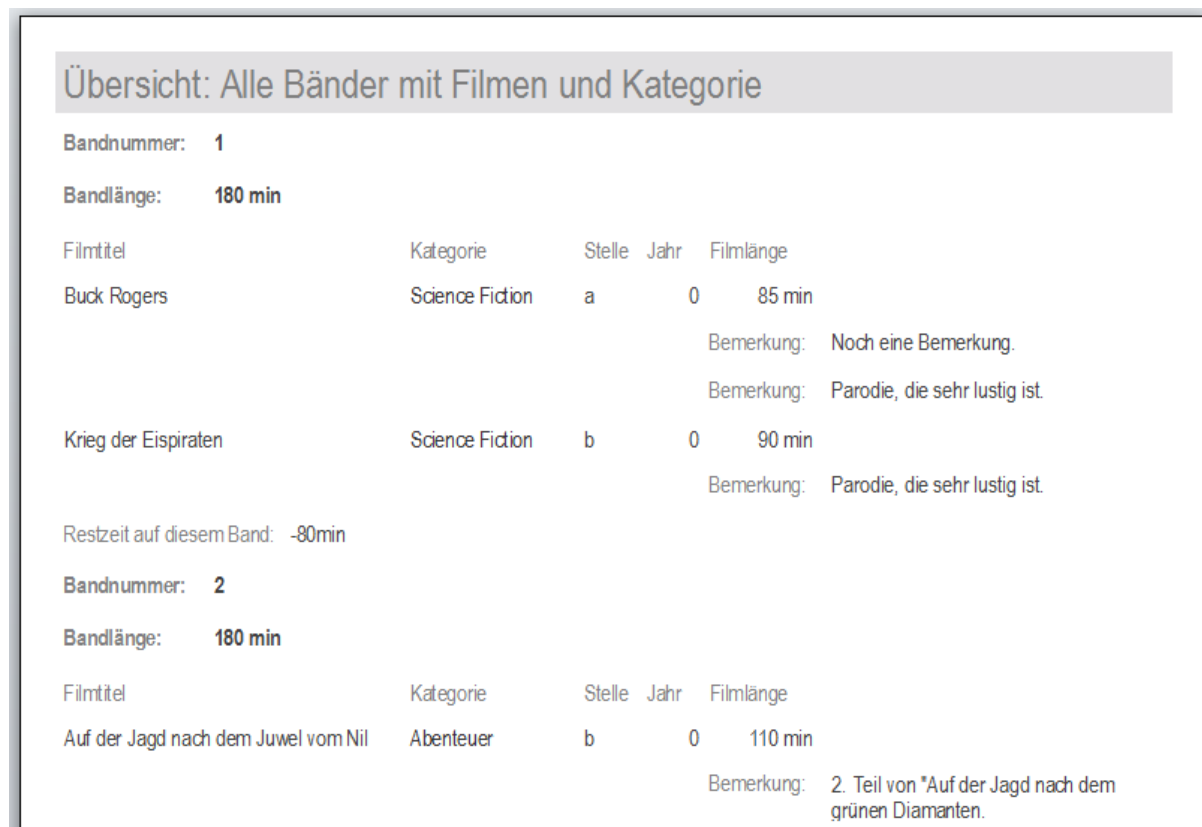




Abd. 6.30 Das neue Feld Restzeit mit Formel

13. Wir wollen hinter der Restzeit auch wieder "min" für Minuten stehen haben, so wie bei der Bandlänge. Das kriegen Sie jetzt sicher auch ohne Anleitung hin.

Das Ergebnis könnte nun in der Seitenansicht so aussehen:



Abd. 6.31 Die Restzeit wird berechnet und im Bericht angezeigt, aber stimmt das Ergebnis?

Das sieht mal wieder ganz schön aus, es gibt jedoch noch ein Problem. Fällt Ihnen beim Ergebnis der Restzeitberechnung etwas auf? Auf dem ersten Band sind ja nur 2 Filme, mit einer Länge von 85 und 90 Minuten, ergibt zusammen insgesamt 175 Minuten. Da das Band 1 eine Länge von 180 Minuten aufweist, beträgt die Restzeit auf dem Band 5 Minuten. Warum kommt dann bei unserer Rechnung die Zahl -80 dabei raus?

Das Problem ist unsere Datenquelle und wird dort auch sofort sichtbar, also in der Abfrage qry\_FilmeBaender. Öffnet man die Abfrage und schaut sich die Zeilen für Band 1 genau an wird deutlich, dass wegen der 2 Kommentare beim Film Buck Rogers auch 2 nahezu identische Zeilen für diesen Film angezeigt werden. Damit rechnet Access in unserem Bericht und damit erklärt sich auch das komische Ergebnis.

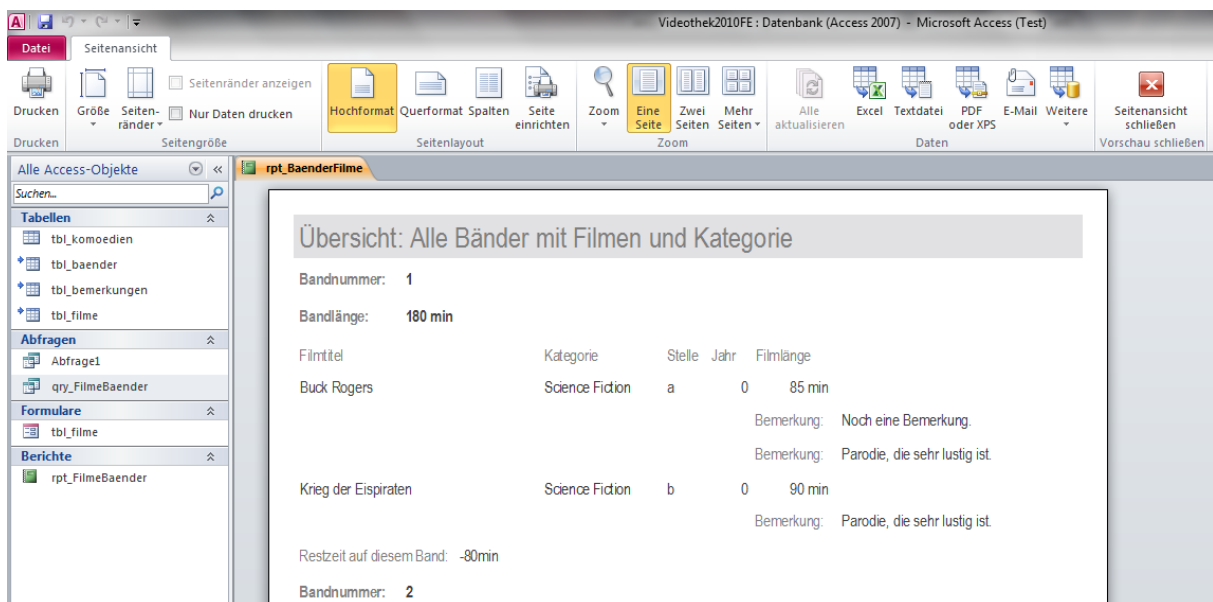


Filmtitel	Kategorie	Bandnumm	Stelle auf dem B	Jahr	Länge	Bandlänge	
Buck Rogers	Science Fiction	1	a	0	85 min	180	Parodie, die sehr lustig ist.
Buck Rogers	Science Fiction	1	a	0	85 min	180	Noch eine Bemerkung.
Krieg der Eispiraten	Science Fiction	1	b	0	90 min	180	Parodie, die sehr lustig ist.
Scottland Yard	Krimi	2	a	0	90 min	180	Dr. Mahuse

Abd. 6.32 Die Resultate der Abfrage qry\_FilmeBaender, die dem Bericht als Datenquelle dient

Die einfachste Lösung für dieses Problem wäre der Verzicht auf das Feld Bemerkung. Wenn man das Feld Bemerkung aus der Abfrage und dem Bericht entfernt und aus der Entwurfsansicht der Abfrage auch die Tabelle tbl\_bemerkungen entfernt, dann stimmen die Ergebnisse. Aber wir hören an dieser Stelle auf. Sie sollten sich selbst noch etwas an den verschiedenen Objektarten versuchen, insbesondere Abfragen. Die Feinheiten beim Umgang mit Formularen und Berichten kommen mit der Zeit (und viel Übung) von ganz alleine.

Einen wirklich ganz kurzen Blick wollen wir zum Abschluss noch auf die Druckeinstellungen und die Seiteneinrichtung werfen. Sämtliche Einstellungen dieser Art lassen sich in der **Seitenansicht** vornehmen. Dort sehen wir auch, wie unser Bericht gedruckt aussehen wird:



Abd. 6.33 Seitenansicht des Berichts

Ganz wichtig ist, dass die Summe von *Links* und *Rechts* und die Breite des Berichts bei einem Hochformat 21 cm, bei einem Querformat 28 cm **nicht** überschreitet. Sollten Sie die Abmessungen einer Seite überschreiten, warnt Sie Access davor. Ob Sie nun die Ränder verkleinern müssen, hängt von ihrer Berichtsbreite ab, die Sie in der Entwurfsansicht über das Lineal verändern können.

So, das war jetzt etwas mehr als ein Zehntel dessen, was dieses Programm leistet. Zur Vertiefung Ihrer Kenntnisse, sollten Sie das neu erlernte Wissen auf jeden Fall so bald wie möglich bei der Erstellung einer neuen oder der Pflege einer bestehenden Datenbank einsetzen. Probieren Sie dabei ruhig auch mal ein paar neue Sachen aus, indem Sie sich bspw. Tipps aus dem Internet holen.