

```
'Diese Funktion spaltet den Inhalt des Felds "AutorName" der Hilfstabelle in Vor-  
und Nachname auf.  
Public Function AutorVorfeldVormameNachnameNormalisieren() As Boolean  
Dim strIDC As String  
Dim strSQL as SQL-Statement  
Dim rstHilfstabelle As DAO.Recordset  
Dim strAutorName As String  
Dim lngLetzteLeereszeichenPosition As Long  
Dim strAutorVormame As String  
Dim strAutorNachname As String  
Do While <not> rstHilfstabelle.EOF  
    'Zunächst öffnen wir DAO-Recordsets zum Ändern der Daten per Programmcode  
    Set dbCurrentDB = CurrentDB()  
    'Bei der Hilfstabelle wird zum Ändern geöffnet.  
    Set rstHilfstabelle = dbCurrentDB.OpenRecordset("tblImportAutorNormalisiert", dbOpenDyn  
    'Die Do-While-Schleife iteriert über alle Datensätze der Hilfstabelle  
    With rstHilfstabelle  
        Do While <not> .EOF = False  
            'Der nicht normalisierte Wert auslesen (Beispiel: "Ralph F. Johnson")  
            strAutorName = "AutorName"  
            'Funktion des letzten Leereszeichens suchen (ergibt in Beispiel den Wert 9)  
            lngLetzteLeereszeichenPosition = InStrRev(strAutorName, " ")  
            'Das letzte Leereszeichen wieder entfernen  
            'Jetzt den Namen in Vorname und Nachname aufsplitten  
            strAutorVormame = Left(strAutorName, lngLetzteLeereszeichenPosition - 1)
```




Wolfram Langer

Access 2019

Das umfassende Handbuch

- ▶ Tabellen, Formulare, Berichte und Abfragen erstellen
- ▶ Datenbankdesign optimieren; Daten filtern, sortieren und auswerten
- ▶ Inkl. Datenimport und -export, SQL, VBA- und DAO-Programmierung, Desktopanwendungen entwickeln, SQL Server als Backend

 Alle Codebeispiele und Beispieldatenbanken zum Download

 Rheinwerk Computing

Kapitel 2

Access als Datenbank: Tabellen

*In einer relationalen Datenbank sind die Daten in Tabellen gespeichert.
Zwischen den Tabellen bestehen Beziehungen.*

Wie der Name schon sagt, dreht sich bei einer Datenbank alles um Daten. Access ist ein Datenbanksystem für relationale Datenbanken. Das bedeutet: Die Daten stehen in Tabellen, die miteinander in Beziehung stehen. In diesem Kapitel dreht sich alles um das Thema »Tabellen«:

- ▶ Tabellen erstellen und ändern
- ▶ Daten eingeben, ändern und löschen

Das ist quasi die Grundausstattung zum Arbeiten mit Access. Darüber hinaus werde ich in diesem Kapitel ausführlich auf das *Datenbankdesign* und auf die *Normalformen* eingehen. Beide Themen beschreiben, wie Sie Daten möglichst optimal in Tabellen ablegen. Dazu kann man sehr abstrakt und theoretisch vorgehen. Ich habe mich bemüht, möglichst nahe an der Praxis zu bleiben. Der eine oder andere Profi in Sachen Datenbanktheorie möge es mir bitte verzeihen, wenn meine Ausführungen an einigen Stellen nicht präzise genug sind. Mir ist das Praktische wichtiger: Wie gelangen Sie mit einem guten Datenbankdesign und den Normalformen zu einer strukturierten Sammlung von Daten?

Ich habe diesbezüglich schon sehr viele Diskussionen zwischen Entwicklern erlebt und viele Datenbanken mit gravierenden Fehlern im Datenbankdesign gesehen. Beides zeigt, dass Datenbankdesign und Normalformen keine einfache Sache sind. Aber keine Angst, ich habe das Kapitel mit vielen Beispielen, einigen Geschichten aus der Praxis und schließlich mit den Wahrheiten, die sich innerhalb der Datenbank-Community bewährt haben (Best Practice), gespickt. Damit werden Sie einen guten Eindruck davon gewinnen, in welche Richtung der Weg zu einer Datenbank frei von Sorgen führt.

2.1 Ein Schnelleinstieg in relationale Datenbanken

Als ersten Einstieg werde ich Ihnen zeigen, wie Sie in einer Desktop-Datenbank eine leere Tabelle erstellen und mit Daten befüllen können. Das Ergebnis ähnelt einer Excel-Tabelle: Eine *unstrukturierte Sammlung von Daten*.

Eine einfache Tabelle (egal, ob in Excel oder Access) – das ist die intuitive Herangehensweise an eine Datenbank. Sie kennen das bestimmt aus der Praxis: die Bestell-Liste, die Liste mit den Lagerbeständen usw.

Vorteil gegenüber einer Excel-Tabelle

Selbst mit einer einfachen Access-Tabelle erreichen Sie schon einen Mehrwert gegenüber Excel. Eine Access-Datenbank kann gleichzeitig von mehreren Benutzern zum Lesen und Schreiben geöffnet sein (*Mehrbenutzerfähigkeit*). Damit entfallen viele der lästigen Telefonate wie »Kannst Du bitte mal aus der Excel-Tabelle gehen?«.

In den Abschnitten zum Datenbankdesign und den Normalformen werde ich Ihnen zeigen, wie Sie systematisch zu einer strukturierten Sammlung von Daten gelangen und welcher Mehrwert damit verbunden ist.

2.1.1 Eine neue Desktop-Datenbank mit Access erstellen

Um zu starten, müssen wir zunächst einmal eine leere Desktop-Datenbank erstellen:

1. Starten Sie Microsoft Access.
2. Klicken Sie auf die Schaltfläche LEERE DATENBANK.



Abbildung 2.1 Gleich nach dem Starten fragt Access, ob Sie mit einer leeren Desktop-Datenbank oder einem der vorgefertigten Beispiele beginnen möchten.

3. Wählen Sie den Speicherort aus, geben Sie einen Namen für die Datenbankdatei an, und klicken Sie auf ERSTELLEN.

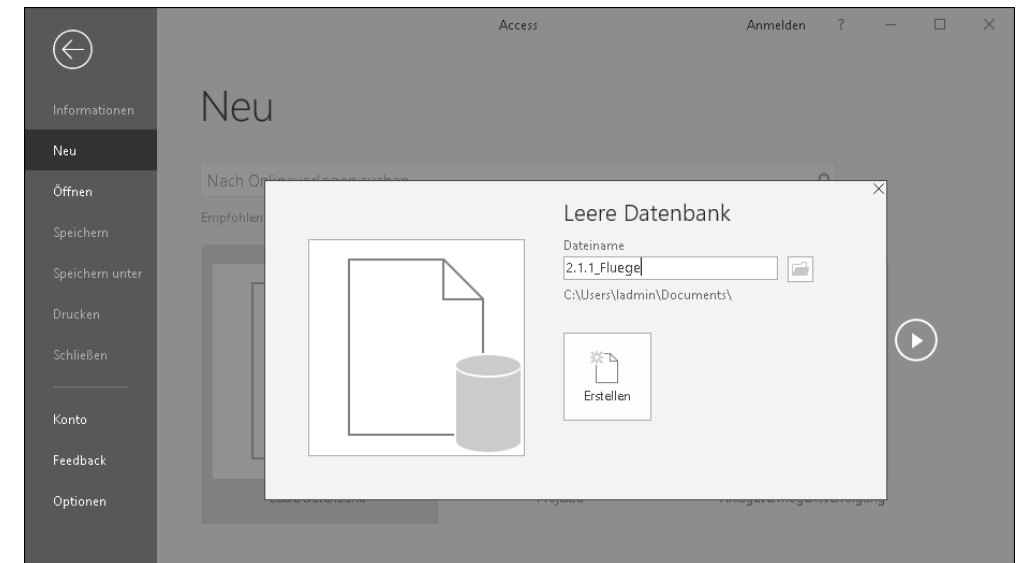


Abbildung 2.2 Geben Sie den Namen und den Speicherort für die ».accdb«-Datei an.

Eine Desktop-Datenbank benötigt immer eine Datei auf der Festplatte

Sie müssen an dieser Stelle einen Dateinamen angeben. Eine Datenbank, die es nur im Arbeitsspeicher gibt (ähnlich einer noch nicht gespeicherten Word- oder Excel-Datei), gibt es in Access nicht. Neben den *.accdb*-Dateien werden übrigens auch noch die älteren *.mdb*-Dateien unterstützt.

Sie gelangen zum Hauptbildschirm von Access (Abbildung 2.3). Auf der linken Seite sehen Sie im *Navigationsbereich* ❶, welche *Datenbankobjekte* in Ihrer Datenbank enthalten sind. Abgesehen von der einen Tabelle ❷ ist die Datenbank erst einmal leer. Später werden Sie im Navigationsbereich die verschiedenen Arten von Datenbankobjekten wiederfinden:

- ▶ Tabellen
- ▶ Abfragen
- ▶ Formulare
- ▶ Berichte
- ▶ Makros
- ▶ VBA-Module

Wir befassen uns in diesem Kapitel nur mit Tabellen.

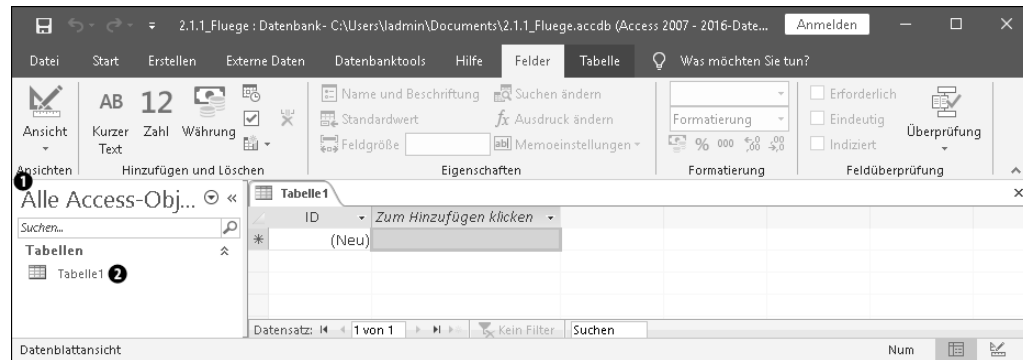


Abbildung 2.3 Access hat in der neuen Datenbank gleich eine leere Tabelle erstellt.

2.1.2 Eine Tabelle in Access erstellen

In der Excel-Tabelle *O2_Access_als_Datenbank_Tabellen\2.1.1_Fluege.xlsx* habe ich einige Datensätze mit Flugbuchungen vorbereitet.

Datum	Abflug	Ankunft	Fluggesellschaft	Flugnummer	Flug von	nach	Passagiere
03.07.2019	15:00	16:20	easyJet	EJU5846	Wien	Berlin-Tegel	
03.07.2019	15:10	16:40	Austrian Airlines	OS131	Wien	Frankfurt am Main	
03.07.2019	15:40	17:20	Air France	AF2334	Paris-Charles-de-Gaulle	Berlin-Tegel	
03.07.2019	15:50	17:35	Air France	AF2035	Berlin-Tegel	Paris-Charles-de-Gaulle	
03.07.2019	15:55	17:15	Air France	AF1919	Frankfurt am Main	Paris-Charles-de-Gaulle	
03.07.2019	15:55	05:35	Lufthansa	LH401	New York John F. Kennedy	Frankfurt am Main	Margot Wilke (Abt. Verkauf)
03.07.2019	16:00	17:10	Lufthansa	LH191	Berlin-Tegel	Frankfurt am Main	Ramona Rathke (Projekt GH)
03.07.2019	16:10	17:15	Lufthansa	LH2890	Berlin-Tegel	Köln/Bonn	Eva Hachmann (Geschäftsführer)
03.07.2019	16:15	16:55	Lufthansa	LH914	Frankfurt am Main	London Heathrow	

Abbildung 2.4 Eine Liste von 50 Flügen mit mehreren Buchungen – das ist eine kleine »Datenbank«, die mit Excel nachempfunden ist.

In ähnlicher Weise können Sie eine Tabelle in Access erstellen:

- Falls Sie die von Access vorbereitete Tabelle versehentlich geschlossen haben: Klicken Sie auf **ERSTELLEN • TABELLEN • TABELLE**.
- Klicken Sie auf **ZUM HINZUFÜGEN KLICKEN**, um ein neues *Feld* (Spalte) einzufügen.
- Im Kontextmenü zeigt Access den *Feldtyp* an (Abbildung 2.5). Wählen Sie der Einfachheit halber den Eintrag **KURZER TEXT** aus.

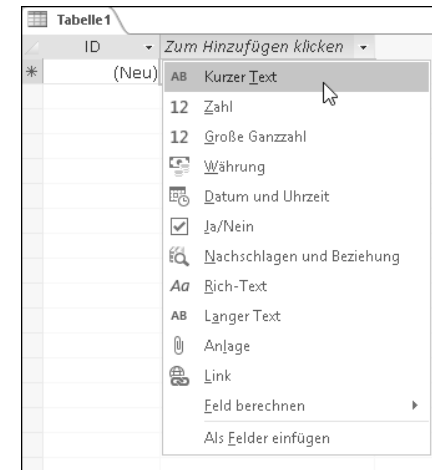


Abbildung 2.5 Jedes Feld hat einen Felddatentyp. Details dazu kommen später; wählen Sie erst einmal »Kurztext« aus.

- Geben Sie den Namen des Feldes an, und wiederholen Sie die Schritte 2–4, um diese Felder zu erstellen:

- »AbflugDatum«
- »AbflugZeit«
- »AnkunftZeit«
- »Fluggesellschaft«
- »Flugnummer«
- »AbflugFlughafen«
- »AnkunftFlughafen«
- »Passagiere«

Übrigens würde ich bei der Benennung von Feldnamen immer auf Leer- und Sonderzeichen verzichten, weil dies später Probleme bereiten könnte. Also verwenden Sie lieber »Flugnummer«, nicht jedoch »Flug-Nr.«.

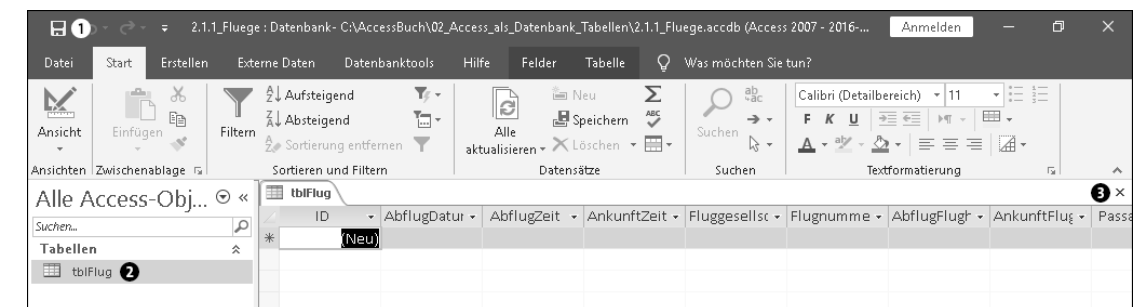





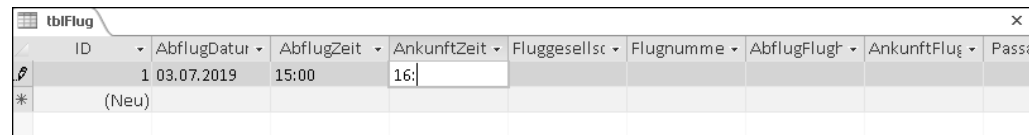
Abbildung 2.6 So sieht die leere Access-Tabelle mit neun Feldern aus.

- Speichern Sie die Tabelle unter dem Namen *tblFlug* ab, indem Sie in der Symbolleiste für den Schnellzugriff auf **SPEICHERN**  klicken (oder **Strg** + **S**).

Im Navigationsbereich links erscheint die neu erstellte Tabelle *tblFlug* . Sie können das Datenblatt schließen  und das Datenbankobjekt per Doppelklick jederzeit wieder öffnen.

2.1.3 Die Tabelle mit Inhalt füllen

Noch ist die Tabelle leer – das soll sich jetzt ändern. Tippen Sie einfach einmal eine Flugbuchung in die erste Zeile.



ID	AbflugDatum	AbflugZeit	AnkomstZeit	Fluggesellschaft	Flugnummer	AbflugFlughafen	AnkomstFlughafen	Passagiere
1	03.07.2019	15:00	16:00	(Neu)				

Abbildung 2.7 Jede Zeile der Tabelle ist ein Datensatz. Sobald Sie mit dem Tippen beginnen, erstellt Access den neuen Datensatz.

Am linken Rand der Zeile erscheint ein kleiner Stift. Access zeigt damit an, dass der Datensatz im *Editiermodus* (*Bearbeitungsmodus*) ist. Sobald Sie in die nächste Zeile gehen, verschwindet der Stift. Erst jetzt wird der Datensatz wirklich in der *.accdb*-Datei gespeichert.

Gemeinsam mit einer Access-Datenbank arbeiten

Sobald der Stift verschwunden ist, ist der Datensatz gespeichert. Die Datenbank selbst muss nicht noch einmal gespeichert werden. Falls andere Personen die Datenbank gleichzeitig geöffnet haben, werden die Änderungen frühestens jetzt für andere sichtbar. Access aktualisiert die Anzeige nach einiger Zeit automatisch (oder aktualisieren Sie sie mit **Strg** + **F9** manuell).

Leider werden neue Datensätze nicht immer automatisch angezeigt. Aber immerhin funktioniert Folgendes zuverlässig:

- ▶ Änderungen an Datensätzen werden sichtbar.
- ▶ Löschungen von Datensätzen werden sichtbar.
- ▶ Access warnt, bevor Änderungen einer anderen Person überschrieben werden.

Damit lässt sich sehr gut gemeinsam an einer Datenbank arbeiten.

Um Tipparbeit zu sparen, können Sie die Excel-Daten bequem über die Zwischenablage in eine Access-Tabelle importieren.

- Öffnen Sie die Excel-Tabelle *O2_Access_als_Datenbank_Tabellen\2.1.1_Fluege.xlsx*.
- Markieren Sie den Bereich mit den Daten, jedoch ohne die Überschriften (Bereich A2:H51).
- Drücken Sie **Strg** + **C**.

- Wechseln Sie zu Access, und öffnen Sie die Tabelle *tblFlug*, falls nicht schon geschehen.
- Markieren Sie in der Kopfzeile die Spalten »AbflugDatum« bis »Passagiere«, und fügen Sie die Datensätze mit **Strg** + **V** ein.
- Access teilt Ihnen mit, dass 50 Datensätze eingefügt werden. Bestätigen Sie den Import, indem Sie auf **JA** klicken.



Abbildung 2.8 Über die Zwischenablage können Sie bequem mehrere Datensätze gleichzeitig aus der Excel-Tabelle importieren.

Noch ein paar Worte zur ersten Spalte, »ID«: Sie ist ein *AutoWert-Feld* und gleichzeitig *Primärschlüssel* der Tabelle *tblFlug*. Access vergibt eine Identifikationsnummer automatisch, die Sie auch nicht selbst festlegen oder ändern können. Wie ich Ihnen noch zeigen werde, ist der Primärschlüssel in einer relationalen Datenbank elementar wichtig.

2.2 Datenbankdesign

Erfahrungsgemäß entstehen neue Datenbankprojekte häufig wie in Abschnitt 2.1, »Ein Schnelleinstieg in relationale Datenbanken«, beschrieben. Und vielleicht kennen Sie das aus dem Arbeitsalltag: Zunächst beginnt alles mit einer Excel-Tabelle, diese wird im Laufe der Zeit immer größer, und irgendwann wird das Ganze recht unübersichtlich. Spätestens wenn mehrere Benutzer mit derselben Excel-Tabelle arbeiten, klingelt permanent das Telefon: »Könntest Du bitte gerade mal die Excel-Tabelle schließen, damit ich die neuen Daten eintragen kann?«, oder: »Kannst Du die Datei für alle zur Bearbeitung freischalten?«. Der Grund für die Anrufe: die gegenseitige Sperrung der Datei. Spätestens zu diesem Zeitpunkt stellt sich heraus, dass die Datenbankmöglichkeiten von Excel ausgereizt sind.

Unabhängig davon, ob bereits die eine oder andere Excel-Tabelle als Datenbankkrücke vorhanden ist, am Anfang eines erfolgreichen Datenbankprojekts steht immer intensive Kopfarbeit. Dies ist die in meinen Augen ehrlich gesagt schwierige und anstrengende Arbeit des sogenannten *Datenbankdesigns*. Beim Prozess des Datenbankdesigns müssen Sie die folgenden Fragen klären:

- ▶ Welche Tabellen sind notwendig?
- ▶ Welche Daten gehören in genau welche Tabelle (Normalisierung der Datenbank)?
- ▶ Können Einschränkungen und Regeln definiert werden, um Ordnung in die Daten zu bringen?
- ▶ In welchen Beziehungen stehen die Tabellen untereinander?

Die Antworten zu diesen Fragen leiten sich aus Ihrem *Geschäftsmodell* ab. Dieser Begriff klingt vielleicht etwas hochtrabend, gerade wenn Sie eine Datenbank für Ihre private Musiksammlung erstellen möchten. Gemeint ist damit die Beschreibung der logischen Funktionsweise in Ihrem Projekt, d. h. wer oder was auf welche Weise miteinander agiert. Einfach gesagt: Worum geht's? Und genau da helfen Ihnen die oben aufgeführten Fragen weiter.

In einem Datenbankprojekt wird die Arbeit des Datenbankdesigns nie vollständig abgeschlossen sein. Für diese wichtige Arbeit nehme ich mir aber gerade beim Start eines neuen Projekts ausreichend Zeit – und genau das empfehle ich Ihnen ebenfalls. Denn Sie werden sehen: Der Aufwand für ein gutes Datenbankdesign zahlt sich im weiteren Projektverlauf für Sie fortwährend aus.

Grundsätzliches zum Datenbankdesign

Diese beiden Punkte sind so wichtig, dass sie einen eigenen Hinweiskasten verdient haben:

- ▶ Stellen Sie sich vor jedem Datenbankprojekt die Fragen aus dem Abschnitt vor dem Kasten, und beantworten Sie sie auch gewissenhaft für sich.
- ▶ Nehmen Sie sich dazu bitte Zeit. Je sorgfältiger Sie planen, desto mehr Zeit und Nerven sparen Sie im Verlauf des Projekts.

Vertrauen Sie wenigstens in diesen beiden Punkten meinem (aus eigener Erfahrung abgeleiteten) Rat. Nicht jede Erfahrung müssen Sie zwangsläufig selbst machen, es sei denn, auch Sie wollen der Kaffeerösterei Ihres Vertrauens durch häufige Nachtschichten zu deutlichen Umsatzsteigerungen verhelfen.

In diesem Abschnitt lassen wir die Excel-Welt hinter uns. Ich zeige Ihnen, wie Sie eine Datenbank mit mehreren Tabellen erstellen, dabei die Übersicht behalten und durch richtiges Datenbankdesign Ordnung in Ihre Datenbank bringen. Sobald Sie den Dreh raus haben, werden Sie feststellen, dass Datenbankdesign richtig Spaß macht und Sie gar nicht mehr aufhören wollen!

2.2.1 Die Welt in Tabellen abbilden

In unserem Beispiel geht es darum, Flugbuchungen in Tabellen festzuhalten. Während der ersten Besprechungen eines neuen Datenbankprojekts frage ich gerne: »Was genau soll

denn gespeichert werden?«, »Was ist wichtig?« und »Was ist unwichtig oder soll erst einmal nicht abgebildet werden?«.

Eine immer wieder gehörte Antwort lautet: »Alles ist wichtig« – aber wenn Sie sich diese Antwort zu eigen machen, dann besteht die Gefahr, dass Sie sich verzetteln.

In unserer Beispieldatenbank sollen nur die folgenden Daten gespeichert werden:

1. Flüge mit Flugnummer, Datum sowie dem Zeitpunkt von Abflug und Ankunft
2. Flughäfen
3. Mitarbeiter, die auf einen oder mehrere Flüge gebucht sind
4. Abteilungen, in denen die Mitarbeiter beschäftigt sind
5. Projekte, an denen die Mitarbeiter beteiligt sind

Dies ist schon eine ganze Menge unterschiedlicher Daten. Grundsätzlich empfehle ich Ihnen, lieber erst einmal mit einem überschaubaren Rahmen von Daten zu starten und Ihre Datenbank danach schrittweise zu erweitern.

In einer relationalen Datenbank versuchen wir, die reale Welt in Tabellen abzubilden. Für jede Art von Gegenstand der realen Welt wird es eine eigene Tabelle geben. In unserem Beispiel werden wir also mindestens fünf Tabellen erstellen:

1. Tabelle *tblFlug*
2. Tabelle *tblFlughafen*
3. Tabelle *tblMitarbeiter*
4. Tabelle *tblAbteilung*
5. Tabelle *tblProjekt*

Anders gesagt: Packen Sie nicht alles in eine Tabelle. Das ist nämlich so ähnlich, als ob Sie in Ihrem Schreibtisch nur eine einzige Schublade nutzen und dort alle möglichen Sachen hineinwerfen. Gut, ich muss zugeben, dass in meinem Regal auch eine »allgemeine Ablage« (eine schöne Umschreibung für »Chaos-Box«) steht. Also eine Zauberkiste, in die ich alle möglichen Schreiben und Notizen werfe und manchmal auch wiederfinde (herauszaubere). Aber ganz ehrlich: Mir persönlich reicht eine dieser Zauberkisten!

Datenbanken sollten das genaue Gegenteil sein: Unterschiedliche Dinge gehören in unterschiedliche Tabellen, das schafft von vornherein Ordnung. Damit Sie und Ihre Kollegen in Bezug auf die Elemente der Datenbank die gleiche Sprache sprechen, kann es an dieser Stelle hilfreich sein, ein Glossar wie in Tabelle 2.1 zu erstellen. Dazu reicht eine einfache Word-Datei, in der jeder Begriff kurz definiert wird. Übrigens finde ich, dass ein Glossar auch dann sehr hilfreich sein kann, wenn man eine Datenbank allein, ohne Projektteam erstellt. Ich habe mich schon häufig dabei ertappt, dass ich Unterschiedliches in eine Tabelle packen wollte. Beim Aktualisieren des Glossars habe ich dann gemerkt, dass ich in Wirklichkeit eine neue Tabelle benötigte.

Begriff	Definition
Flug	Transport von Personen per Flugzeug von einem <i>Flughafen</i> zu einem anderen. Jeder Flug hat eine <i>Flugnummer</i> , die zusammen mit dem <i>Datum</i> eindeutig ist.
Flughafen	Örtlicher Start- und Endpunkt eines <i>Fluges</i> . In einer Stadt kann es auch mehrere Flughäfen geben. Jeder Flughafen hat einen eindeutigen Namen und einen eindeutigen <i>IATA-Code</i> . Die Liste der Flughäfen wird durch die IATA gepflegt (http://en.wikipedia.org/wiki/List_of_airports).
Flugnummer	Umgangssprachliche Bezeichnung für <i>flight code</i> gemäß der IATA-Definition. Besteht aus dem IATA-Code für die Fluggesellschaft (http://en.wikipedia.org/wiki/List_of_airlines) und einer Zahl (<i>flight number</i>). Beispiel: LH401
IATA	<i>International Air Transport Association</i> , der internationale Dachverband der Fluggesellschaften
...	...

Tabelle 2.1 Wichtige Begriffe sollten Sie in einem Glossar definieren, damit alle im Projektteam die gleiche Sprache sprechen.

In Abschnitt 2.8, »Durch die Normalformen das Datenbankdesign verbessern«, werde ich Ihnen zeigen, wie Sie Daten systematisch in unterschiedliche Tabellen trennen und welche Vorteile Sie dadurch erzielen. In Abschnitt 2.8.8, »Zweckmäßiges Datenbankdesign«, zeige ich Ihnen, wie Sie dabei möglichst nahe an der Realität bleiben und sich dadurch gleichzeitig Dokumentationsarbeit ersparen können.

Best Practice beim Abbilden der Welt in Tabellen

- ▶ Definieren Sie wichtige Begriffe (Glossar).
- ▶ Aktualisieren Sie das Glossar im Projektverlauf.
- ▶ Unterschiedliche Sachen gehören auch in unterschiedliche Tabellen.
- ▶ Bleiben Sie möglichst nahe an der Realität.

2.2.2 Namenskonventionen

Bevor wir die Tabellen erstellen, eine scheinbar ganz banale Frage: Wie sollen die Tabellen überhaupt heißen? Sobald Sie mehr als eine Tabelle haben, ist diese Frage überhaupt nicht

mehr trivial. Ich mache mir im Gegenteil jedes Mal sehr viele Gedanken darüber, wie ich eine Tabelle oder ein Feld treffend bezeichne und ob die Benennung den Inhalt der Tabelle auf den Punkt bringt.

Das lässt Access bei der Benennung technisch zu

In Access gibt es von technischer Seite nur sehr wenige Einschränkungen bei der Benennung von Tabellen und Feldern:

- ▶ maximal 64 Zeichen
- ▶ alle Zeichen mit Ausnahme von eckigen Klammern (»[« und »]«), Punkt (».«) und Ausrufezeichen (»!«)
- ▶ Leerzeichen nicht am Anfang oder am Ende

Rein technisch gesehen haben Sie also recht freie Wahl. Sie dürfen auch Wörter verwenden, die bei der Programmierung als reservierte Schlüsselwörter gelten (beispielsweise in der Abfragesprache SQL oder der Programmiersprache Visual Basic for Applications).

Mit diesen Regeln der Benennung behalten Sie den Überblick

Trotz aller Freiheiten möchte ich Ihnen gerne einige freiwillige Einschränkungen bei der Benennung ans Herz legen, die sich in der Praxis bewährt haben.

1. Verwenden Sie im Namen von Tabellen (und anderen Datenbankobjekten) oder Feldern *keine Umlaute, Sonderzeichen oder Leerzeichen*.

Access erscheint mir recht fehlerfrei beim Umgang mit exotischen Zeichen zu sein. Das gilt aber nicht unbedingt für Erweiterungen, Programmbibliotheken, Werkzeuge etc. Solche Software ist mitunter nur mit einem englischen Windows und Access entwickelt oder getestet worden, so dass es mit deutschen Umlauten und Sonderzeichen Probleme geben kann. Zu dieser Art von Fehlern werden Sie im Internet auch nur begrenzt Informationen finden, denn die große Anzahl der englischsprachigen Access-Anwender wird nie auf diese Fehler stoßen.

Diese ärgerliche und, wie ich selbst erleben musste, auch ziemlich frustrierende Fehlersuche möchte ich Ihnen gerne ersparen. Verwenden Sie daher ausschließlich die Groß- und Kleinbuchstaben von A bis Z, Zahlen und wenn notwendig den Unterstrich (»_«).

2. Sie erleichtern sich die Arbeit bei der Benennung, indem Sie *stets die Einzahl verwenden*. Also »Flug« als Tabellename, nicht »Flüge«. Zum einen vermeiden Sie von vornherein die Gefahr, einen Umlaut einzutippen, denn viele Pluralwörter haben in der deutschen Sprache mindestens einen Umlaut! Zweitens steht in einer Datenbanktabelle in den seltensten Fällen wirklich nur ein einziger Datensatz. Daher ist es nicht sinnvoll, im Tabellennamen nach der Einzahl oder der Mehrzahl zu unterscheiden. Verwenden Sie stattdessen die kürzere Singularform.

3. Manchmal reicht ein einziges Wort nicht aus, um den Inhalt einer Tabelle oder eines Feldes treffend zu beschreiben. Denken Sie einmal an eine Tabelle, in der eingetragen wird, welcher Mitarbeiter von wann bis wann in einer bestimmten Abteilung des Unternehmens tätig war. Bitte verwenden Sie *bei zusammengesetzten Wörtern* keine Leerzeichen, auch wenn das technisch möglich wäre. Besser ist es, wenn Sie entweder *den Unterstrich* (»Mitarbeiter_tätig_in_Abteilung«) oder die sogenannte *CamelCase-Schreibweise* verwenden, also immer den ersten Buchstaben eines Wortes großschreiben (»MitarbeiterTätigInAbteilung«). Da ich ein Befürworter von wenig Schreibarbeit und damit von wenigen Zeichen bin, verwende ich den CamelCase-Stil.
4. In der Access-Welt hat sich die Benennung nach der sogenannten *Leszynski Naming Convention* (LNC) – einer Variante der sogenannten *ungarischen Notation* – eingebürgert. Nach dieser Konvention bekommt jeder Name ein *Präfix*, das sich aus dem Typ ableitet. Eine Tabelle wird nach der LNC beispielsweise *tblFlug* genannt. Der Vorteil liegt auf der Hand: Sie erkennen anhand des Präfixes sofort, ob Sie eine Tabelle (*tbl*), eine Abfrage (*qry*) oder etwas ganz anderes vor sich haben. Das erhöht die Verständlichkeit Ihrer Datenbank ungemein! Tabelle 2.2 gibt Ihnen einen Überblick über die Präfixe für die gängigsten Datenbankobjekte.

Datenbankobjekt	englische Bezeichnung	LNC-Präfix
Tabelle	<i>table</i>	<i>tbl</i>
Abfrage	<i>query</i>	<i>qry</i>
Formular	<i>form</i>	<i>frm</i>
Bericht	<i>report</i>	<i>rpt</i>
Makro	<i>macro</i>	<i>mcr</i>
Modul	<i>module</i>	<i>bas</i> (angelehnt an »basic module«)
Klassenmodul	<i>class module</i>	<i>cls</i>

Tabelle 2.2 Präfixe für Datenbankobjekte nach der Leszynski Naming Convention (LNC)

Neben diesen sieben Präfixen gibt es weitere, unter anderem für verschiedene Arten von Tabellen oder Abfragen. Ob eine solche feinere Unterteilung nützlich ist, mag Geschmackssache sein. Ich begnüge mich mit *tbl* für Tabellen und *qry* für Abfragen.

Auch für die Bezeichnung von Feldern nach Felddatentyp wurden Präfixe in der LNC festgelegt. Mit Felddatentypen werden wir uns ausführlich in Abschnitt 2.3, »Erstellen von Tabellen«, beschäftigen. In Bezug auf eine Namenskonvention gibt es das Problem, dass sich Felddatentypen zwischen Access, SQL und anderen Datenbanksystemen allein vom Namen her schon unterscheiden und es daher leicht zu Verwirrungen kommt. Von LNC-Präfixen für Feldnamen rate ich Ihnen daher ab.

Felddatentyp	englische Bezeichnung	Jet-SQL-Datentyp	LNC-Präfix
AUFWERT ZUFALL	<i>AutoNumber Random</i>	–	<i>idn</i>
AUFWERT REPLIKATIONS-ID	<i>AutoNumber Replication ID</i>	–	<i>idr</i>
AUFWERT INKREMENT	<i>AutoNumber Increment</i>	COUNTER	<i>ids</i>
BINÄR	<i>Binary</i>	BINARY	<i>bin</i>
BYTE	<i>Byte</i>	TINYINT	<i>byt</i>
WÄHRUNG	<i>Currency</i>	MONEY	<i>cur</i>
DATUM/UHRZEIT	<i>Date/Time</i>	DATETIME	<i>dtm</i>
DOUBLE	<i>Double</i>	FLOAT	<i>dbl</i>
LINK	<i>Hyperlink</i>	–	<i>hlk</i>
INTEGER	<i>Integer</i>	SMALLINT	<i>int</i>
LONG INTEGER	<i>Long</i>	INTEGER	<i>lng</i>
LANGER TEXT	<i>Long Text</i>	TEXT	<i>mem</i> (angelehnt an »memo«)
OLE-OBJEKT	<i>OLE Object</i>	IMAGE	<i>ole</i>
SINGLE	<i>Single</i>	REAL	<i>sng</i>
KURZER TEXT	<i>Short Text</i>	CHAR VARCHAR	<i>chr</i> (angelehnt an »character«)
JA/NEIN	<i>Yes/No (Boolean)</i>	BIT	<i>ysn</i>

Tabelle 2.3 Falls Sie wirklich LNC-Präfixe für Feldnamen verwenden wollen, beachten Sie bitte, dass z. B. das Präfix »int« nicht dem SQL-Datentyp »INTEGER« entspricht!

Eine andere Herangehensweise ist, die Felder mit einem thematischen Präfix zu versehen. Gemeint ist eine *eindeutige* Abkürzung wie beispielsweise »flg« für die Tabelle *tblFlug*. Die Felder heißen dann »flgID«, »flgAbflugDatum«, »flgAbflugZeit« ... Ein wesentlicher Vorteil liegt auf der Hand: Jeder Feldname ist nun eindeutig in der gesamten Datenbank, Sie können schon anhand des Präfixes die Tabelle erkennen oder erraten. Ich könnte mir vorstellen, dass diese Konvention für kleinere Datenbanken praktikabel ist. Am besten fügen Sie eine zusätzliche Spalte in Ihrer Word-Datei des Glossars ein, dann haben Sie das Präfix

immer im Blick. Achten Sie in jedem Fall darauf, dass jedes Präfix eindeutig für eine Tabelle ist, denn sonst ist das Konzept der thematischen Präfixe für Feldnamen sinnlos. Sobald Ihre Datenbank mehrere Tabellen enthält, kann es durchaus schwierig werden, geeignete und gleichzeitig eindeutige Präfixe zu finden. Ich verwende thematische Präfixe aus genau diesem Grund nicht.

Sie haben sicherlich gemerkt, dass es bei den Namenskonventionen unterschiedliche Meinungen und Herangehensweisen gibt. Ich empfinde Präfixe bei der Bezeichnung von Feldern als wenig sinnvoll, jedoch *tbl* für Tabellen und *qry* für Abfragen als sehr hilfreich. Wie Sie sich auch immer entscheiden: Legen Sie sich zu Projektbeginn auf eine einheitliche Richtlinie zur Benennung von Tabellen und Feldern fest. Das ist umso wichtiger, wenn Sie in einem Team arbeiten.

Best Practice bei der Benennung von Tabellen und Feldern

- ▶ *tbl* als Präfix für den Tabellennamen
- ▶ keine Umlaute
- ▶ keine Sonderzeichen
- ▶ keine Leerzeichen
- ▶ maximal 64 Zeichen
- ▶ einheitlich CamelCase oder den Unterstrich verwenden
- ▶ stets die Einzahl verwenden

Unter Berücksichtigung dieser Namenskonventionen sieht die Tabelle *tblFlug* nun so aus:

tblFlug
FlugID
AbflugDatum
AbflugZeit
AnkunftsZeit
Fluggesellschaft
Flugnummer
AbflugFlughafen
AnkunftFlughafen
Passagiere

Abbildung 2.9 Die Tabelle »tblFlug« unter Einhaltung der Namenskonventionen

Ihnen ist vielleicht aufgefallen, dass ich das erste Feld in »FlugID« umbenannt habe. Dieses Feld ist der sogenannte *Primärschlüssel*, ein äußerst wichtiges Element, das ich Ihnen im folgenden Abschnitt vorstellen möchte.

2.2.3 Der Primärschlüssel

Jede Tabelle sollte einen *Primärschlüssel* (englisch *primary key*) haben. Er ist eindeutig für einen Datensatz, das heißt, wenn Sie den Primärschlüssel kennen, finden Sie auch den Datensatz wieder. Hinzu kommt, dass die Reihenfolge der Datensätze in einer Datenbanktabelle nicht festgelegt ist! Der Primärschlüssel ist entweder genau ein Feld der Tabelle oder die Kombination mehrerer Felder. Wichtig ist nur:

- ▶ Der Primärschlüssel ist *eindeutig*.
- ▶ Der Primärschlüssel *ändert sich nie*.

AutoWert-Feld als Primärschlüssel

Das Feld »FlugID« in Abbildung 2.9 ist der Primärschlüssel von *tblFlug*. Hierbei handelt es sich um ein *AutoWert-Feld*, das heißt um einen Zähler, mit dem Access automatisch alle Datensätze nummeriert. »FlugID« ist also eine künstlich vergebene Identifikationsnummer (ID), die keinen Bezug zur Realität hat.

Felder mit realen Inhalten als Primärschlüssel

Aber auch Felder mit realen Inhalten können Primärschlüssel sein. In den meisten Tabellen werden Sie ein oder mehrere Felder finden, die eindeutige Werte haben und sich daher prinzipiell als Primärschlüssel eignen. Man nennt sie *Schlüsselkandidaten*. Beispielsweise ist in unserer Tabelle *tblFlug* die Kombination der Felder »AbflugDatum« und »Flugnummer« eindeutig und daher ein Schlüsselkandidat. Es ist nicht ungewöhnlich, wenn Sie in einer Tabelle mehrere Schlüsselkandidaten finden. Ich empfehle Ihnen, möglichst jeden Schlüsselkandidaten wie in Abschnitt 2.9.2, »Eindeutiger Schlüssel (Alternativschlüssel)«, erläutert als *Alternativschlüssel* zu definieren.

Feld oder Kombination von Feldern	Schlüsselkandidat	Primärschlüssel	Alternativschlüssel	stellvertretender Schlüssel (»surrogate key«)	natürlicher Schlüssel (»natural key«)
»FlugID«	•	•		•	
»AbflugDatum« »Flugnummer«	•		•		•
»AbflugDatum« »AbflugZeit« »Fluggesellschaft« »AbflugFlughafen« »AnkunftFlughafen«	•		•		•

Tabelle 2.4 Schlüsselkandidaten in der Tabelle »tblFlug«

Feld oder Kombination von Feldern	Schlüsselkandidat	Primärschlüssel	Alternativschlüssel	stellvertreter-Schlüssel (»surrogate key«)	natürlicher Schlüssel (»natural key«)
»AbflugDatum« »AnkunftZeit« »Fluggesellschaft« »AbflugFlughafen« »AnkunftFlughafen«	•		•		•
»AbflugDatum« »AbflugZeit«	kein Schlüsselkandidat, denn zwei Flüge können zeitgleich an unterschiedlichen Flughäfen starten				
»AbflugDatum« »AbflugFlughafen« »AbflugZeit«	ebenfalls <i>kein</i> Schlüsselkandidat, denn ein Flughafen kann zwei Startbahnen haben, auf denen zeitgleich Flugzeuge starten				

Tabelle 2.4 Schlüsselkandidaten in der Tabelle »tblFlug« (Forts.)

Eignet sich nun jeder Schlüsselkandidat als Primärschlüssel? Nein, denn es gibt noch eine weitere Eigenschaft des Primärschlüssels, die gerne vergessen wird: Der Primärschlüssel eines Datensatzes darf sich *nie* ändern. An dieser Einschränkung scheitern leider die meisten Schlüsselkandidaten. Irgendwann meldet sich beispielsweise eine Sachbearbeiterin bei Ihnen und sagt, sie habe sich bei der Flugnummer vertippt. Und schon haben Sie das Dilemma.

Künstlich vergebene IDs als Primärschlüssel helfen, Probleme zu vermeiden!

Mit künstlich vergebenen IDs als Primärschlüssel können Sie das Problem von vornherein umgehen. Diese Art von Primärschlüsseln hat sich in der Praxis derart bewährt, dass sie sogar einen eigenen Namen bekommen haben: *stellvertretende Schlüssel* (englisch *surrogate key*). In Abgrenzung zu den künstlichen vergebenen IDs bezeichnet man die anderen Schlüsselkandidaten übrigens als *natürliche Schlüssel* (englisch *natural key*).

Best Practice zum Primärschlüssel

- ▶ Jede Tabelle bekommt einen Primärschlüssel.
- ▶ Legen Sie ein AUTOWERT-Feld mit künstlich vergebenen IDs als Primärschlüssel fest.
- ▶ Lehnen Sie die Benennung an den Tabellennamen an (»tblFlug« und Primärschlüssel »FlugID«)

2.2.4 Die Tabellenstruktur festlegen

In Abschnitt 2.2.1, »Die Welt in Tabellen abbilden«, haben wir uns ausführlich angesehen, welche Tabellen für unser Beispiel, die Flug-Datenbank, notwendig sind und wie wir sie benennen werden. In diesem Abschnitt werden wir die *Tabellenstruktur* festlegen. Tabellen in einer relationalen Datenbank (zu denen auch Access-Datenbanken gehören) unterscheiden sich in ein paar Merkmalen wesentlich von Excel-Tabellen:

Merkmal	Excel (ab Version 2007)	relationale Datenbank (Access)
Anzahl der Spalten	maximal 16.384	maximal 255
Bezeichnung der Felder (Spalten)	A, B, C ... (Spaltennamen werden üblicherweise in Zeile 1 eingetragen.)	Muss festgelegt werden.
Felddatentyp	variabel (kann individuell für jede Zelle festgelegt werden)	Muss für jede Spalte festgelegt werden.
Anzahl der Datensätze (Zeilen)	maximal 1.048.576	Unbegrenzt! (für eine Access-Datenbank limitiert durch maximal 2 GB für die Größe einer Datenbankdatei)
Sortierung der Datensätze (Zeilen)	Die Sortierung der Zeilen bleibt unverändert. Über SORTIEREN können Sie neu sortieren.	Die Sortierung der Zeilen kann sich ändern! Sortierte Datensätze können Sie aber jederzeit durch eine Abfrage erhalten.
Regeln zur Einschränkung	Datenüberprüfung (Gültigkeitskriterien können individuell für jede Zelle festgelegt werden.)	▶ Feldeinschränkungen ▶ Tabelleneinschränkungen
Mehrere Anwender können gleichzeitig mit einer Tabelle arbeiten.	nein	ja

Tabelle 2.5 Wenn Sie Flexibilität benötigen, ist Excel Ihr Werkzeug der Wahl. Mit relationalen Datenbanken schaffen Sie Ordnung – selbst bei extrem großen Datenmengen.

Vor dem Füllen mit Daten müssen Sie die Tabellenstruktur festlegen. Erst strukturieren, dann mit Daten füllen – das klingt zunächst unnötig kompliziert. Schließlich geht es doch auch anders, wie Sie es vielleicht von Excel gewohnt sind. An dieser Stelle möchte ich Sie noch einmal an Abschnitt 2.2.1, »Die Welt in Tabellen abbilden«, erinnern: Es ist *die* grundlegende Idee von Datenbanken, von vornherein für Ordnung zu sorgen. Das heißt, Daten zu trennen und klare Regeln für die Daten (was kommt wohin?) aufzustellen.

Mit diesen Schritten legen Sie die Tabellenstruktur fest

Nachdem Sie festgelegt haben, in welchen Tabellen Ihre Daten gespeichert werden sollen, müssen Sie für jede einzelne Tabelle die Tabellenstruktur festlegen. Dies umfasst jeweils die folgenden drei Schritte:

1. die Felder anlegen
2. für jedes Feld den Felddatentyp festlegen
3. Regeln zur Einschränkung festlegen

Auch diese Schritte sind sehr wichtig und gehören zum Prozess des Datenbankdesigns.

2.3 Erstellen von Tabellen

Beim Datenbankdesign habe ich Ihnen gezeigt, vor welchen Aufgaben wir beim Erstellen einer relationalen Datenbank stehen:

- ▶ Tabellen festlegen
- ▶ Tabellenstruktur festlegen (Felder mit Felddatentyp und Einschränkungen anlegen)
- ▶ Tabellen und Felder benennen

All das lässt sich auf dem Papier erledigen. Und vielleicht werden Sie staunen: Bei komplizierten Datenbankprojekten setze ich mich wirklich erst einmal mit Papier, Bleistift und Radiergummi (sehr wichtig!) hin.

In den nächsten Abschnitten werde ich Ihnen zeigen, wie Sie das Ergebnis, das Datenbankdesign, in Access übertragen. Und selbstverständlich dürfen Sie auch direkt in Access beginnen und das Datenbankdesign am Bildschirm aufbauen!

2.3.1 Die Entwurfsansicht

In Access können Sie eine Tabelle in zwei Ansichten öffnen:

1. in der *Datenblattansicht*
2. in der *Entwurfsansicht*

Durch Doppelklick auf eine bestehende Tabelle wird immer die Datenblattansicht geöffnet. Mit der Datenblattansicht werden wir uns ausführlich in Abschnitt 2.5, »Daten in Tabellen«, beschäftigen. Wenn Sie die Tabellenstruktur ändern möchten, müssen Sie die Tabelle in der Entwurfsansicht öffnen.

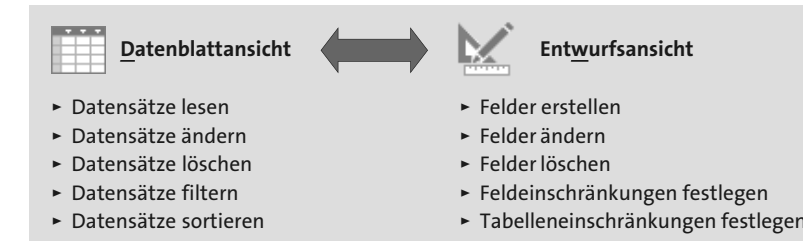


Abbildung 2.10 Verwenden Sie die Datenblattansicht, um die Daten in einer Tabelle zu ändern. Wechseln Sie in die Entwurfsansicht, um die Tabellenstruktur zu verändern.

So erstellen Sie eine neue Tabelle in der Entwurfsansicht

Eine neue Tabelle ohne Felder erstellen Sie wie folgt (Abbildung 2.11):

1. Gehen Sie zu **ERSTELLEN** ❶ • **TABELLEN**.
2. Klicken Sie auf **TABELLENENTWURF** ❷.

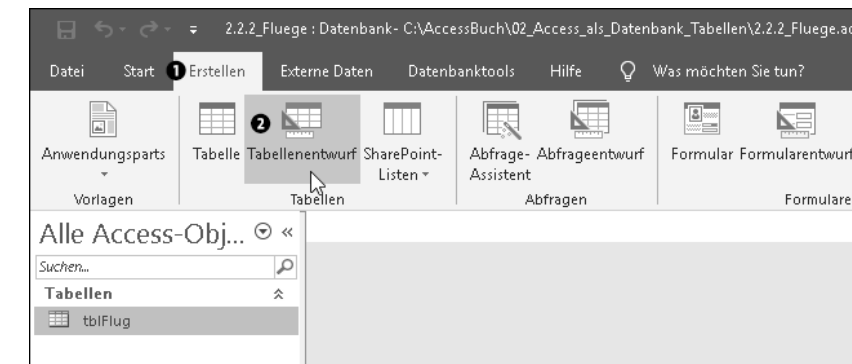


Abbildung 2.11 Eine neue Tabelle in der Entwurfsansicht erstellen

Access öffnet die leere Tabelle in der Entwurfsansicht.

So öffnen Sie eine bestehende Tabelle in der Entwurfsansicht

Wenn Sie die Struktur einer bestehenden Tabelle verändern möchten, öffnen Sie die Tabelle in der Entwurfsansicht (Abbildung 2.12):

1. Gehen Sie im *Navigationsbereich* ❶ auf die Tabelle ❷.
2. Klicken Sie mit der rechten Maustaste.
3. Klicken Sie im Kontextmenü auf **ENTWURFSANSICHT** ❸ (oder **Strg** + **↵**).

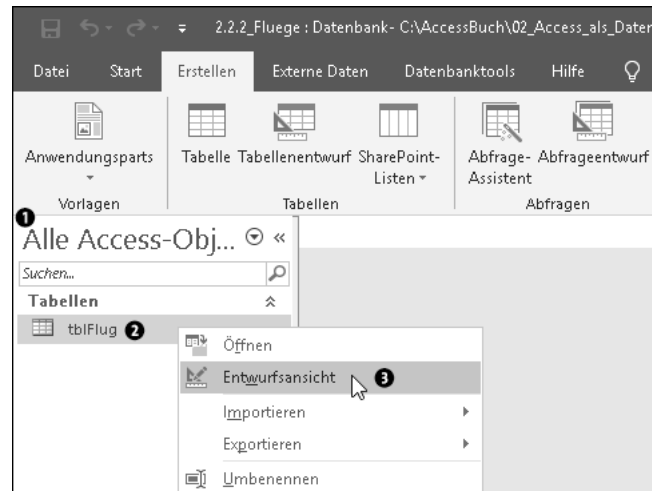


Abbildung 2.12 Eine bestehende Tabelle in der Entwurfsansicht öffnen

Sie können das einmal an der fertigen Beispieldatenbank ausprobieren, die Sie in den Materialien zum Buch unter `O2_Access_als_Datenbank_Tabellen\2.7.5_Fluege_umfassend.accdb` finden.

So schalten Sie eine geöffnete Tabelle von der Datenblattansicht in die Entwurfsansicht um

Ebenso können Sie von der Datenblattansicht ausgehend in die Entwurfsansicht umschalten:

1. Gehen Sie zu **START** ① • **ANSICHTEN** • **ANSICHT** ②.
2. Klicken Sie auf **ENTWURFSANSICHT** ③ (oder `Strg` + `]`).

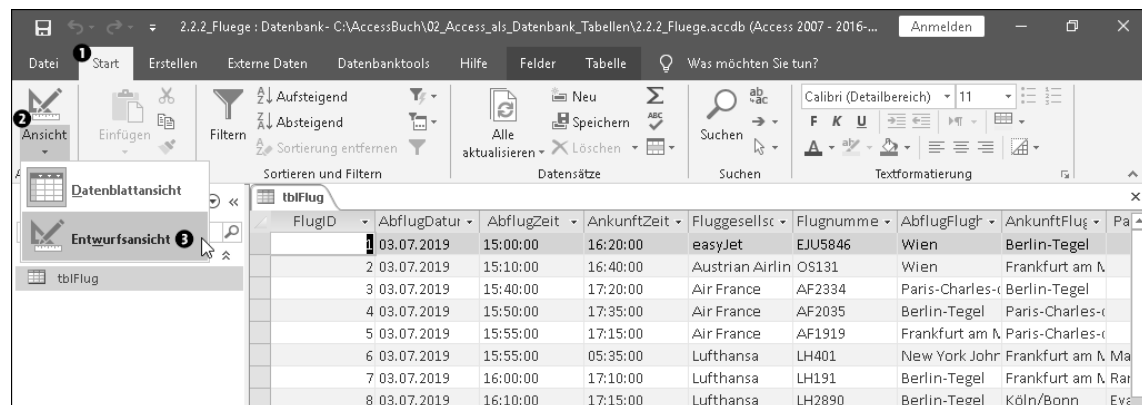


Abbildung 2.13 Aus der Datenblattansicht in die Entwurfsansicht umschalten

So speichern Sie Änderungen an der Tabellenstruktur ab

In der Entwurfsansicht können Sie Änderungen an der Tabellenstruktur durchführen. Die Entwurfsansicht ist der einzige Ort in Access, bei dem Änderungen nicht direkt in der `.accdb`-Datei gespeichert werden. Erst wenn Sie in der Symbolleiste für den Schnellzugriff auf **SPEICHERN** (oder `Strg` + `S`) klicken, wird die neue Tabellenstruktur wirklich gespeichert.

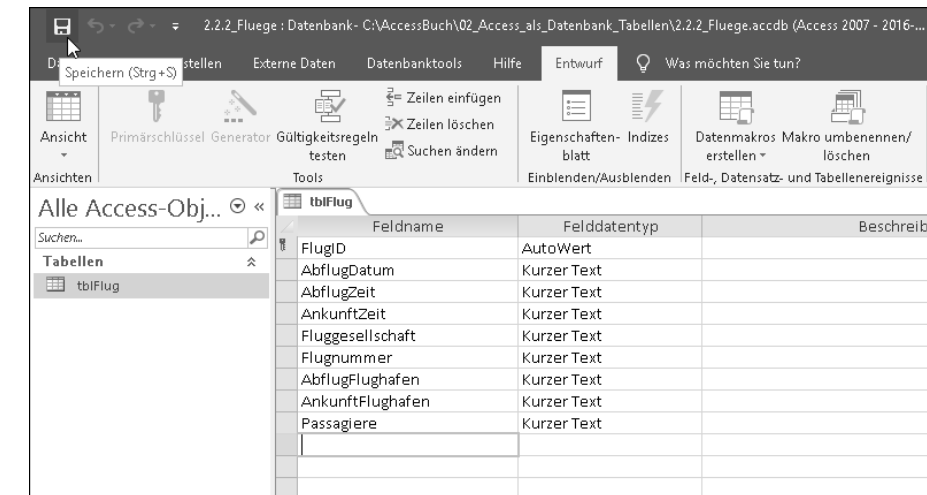


Abbildung 2.14 Speichern Sie Änderungen an der Tabellenstruktur ab, indem Sie auf das Diskettensymbol klicken.

Änderung einer Tabelle mit Datensätzen: Automatische Datenüberprüfung und -anpassung

Sollte Ihre Tabelle Datensätze enthalten, wird Access diese vor dem Speichern überprüfen. Auf dieses Thema – die automatische Überprüfung und Anpassung von Daten – werde ich in Abschnitt 2.4, »Ändern von Tabellen«, genauer eingehen. Sie wissen jetzt aber bereits, dass es so etwas wie eine automatische Datenüberprüfung und -anpassung in Access überhaupt gibt.

Am besten fangen Sie erst einmal ohne Datensätze an. Dann lässt sich die Tabellenstruktur ohne Rücksicht auf Inhalte jederzeit verändern.

Ein Feld in der Entwurfsansicht erstellen

In der Entwurfsansicht sind die Felder einer Tabelle untereinander dargestellt, und so tragen Sie ein neues Feld ein (Abbildung 2.15):

1. Tippen Sie zunächst den Namen des Feldes unter **FELDDATENTYP** ① ein.
2. Anschließend wählen Sie rechts daneben unter **FELDDATENTYP** ② den passenden Felddatentyp aus.

3. Weitere Eigenschaften zum Felddatentyp, das FORMAT, das EINGABEFORMAT und einige weitere Feldeinschränkungen und -eigenschaften können Sie im Registerblatt ALLGEMEIN **3** einstellen.

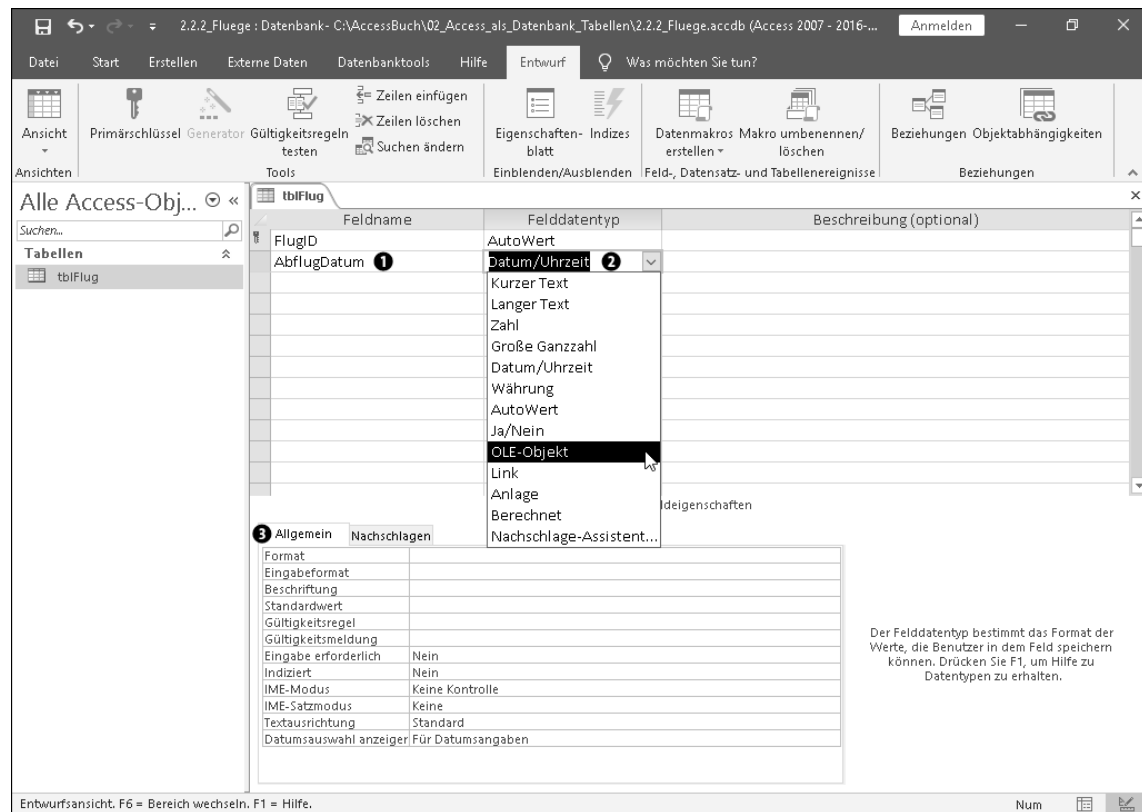


Abbildung 2.15 In der Entwurfsansicht legen Sie die Felder einer Tabelle mit ihren Felddatentypen fest.

Warum ist es wichtig, den passenden Felddatentyp zu wählen?

Mit dem Felddatentyp legen Sie fest, welche Art von Werten Access für ein Feld in der Tabelle überhaupt zulässt. Sie haben mittlerweile bestimmt gemerkt, dass ich Ordnung sehr liebe ... Genau die erreichen Sie mit Felddatentypen: Wenn Sie als Felddatentyp beispielsweise DATUM/UHRZEIT wählen, sorgt Access dafür, dass im Feld »AbflugDatum« wirklich nur ein Datum steht. Im Fachjargon sagt man, dass Access die *Datenkonsistenz* sicherstellt.

Es gibt aber weitere Gründe, warum es lohnt, nach dem passenden Felddatentyp zu suchen:

- ▶ Mit Felddatentypen können Sie Speicherplatz sparen.
- ▶ Sie können Inhalte richtig filtern und sortieren.
- ▶ Sie können Inhalte unterschiedlich formatieren.

Speicherplatz ist bei den heutigen Festplattenkapazitäten und -preisen eher selten ein Problem. Bei Datenbanken dürfen Sie aber nicht vergessen: Die Masse (= die große Anzahl der Datensätze) macht's!

Filtern und Sortieren funktioniert überhaupt nur richtig, wenn Sie die passenden Felddatentypen verwenden. Sie kennen das: Es ist ein großer Unterschied, ob Sie Werte als Text, als Zahl oder als Datum sortieren.

Unterschiedliche *Formatierungen* sind besonders bei Datum und Uhrzeit interessant.

Alle drei Punkte sind also sehr wichtig. Daher empfehle ich Ihnen:

- ▶ Machen Sie sich mit den einzelnen Felddatentypen vertraut.
- ▶ Nehmen Sie sich Zeit für die Suche nach dem passenden Felddatentyp.
- ▶ Verwenden Sie denjenigen Felddatentyp, der die Realität bestmöglich abbildet.

In den nächsten Abschnitten werde ich Ihnen die einzelnen Felddatentypen ausführlich vorstellen. Auch die anderen Einstellungen im Registerblatt ALLGEMEIN werde ich Ihnen im Detail erläutern.

Wenn Sie nach dem Abspeichern der Tabellenstruktur in die Datenblattansicht umschalten, werden alle Felder als Spalten dargestellt. Access dreht also die Anordnung der Felder zwischen beiden Ansichten um 90°. Um Verwirrung zu vermeiden, werde ich mich bemühen, nicht von Zeilen oder Spalten, sondern nur von *Datensätzen* und *Feldern* zu sprechen.

Änderungen der Tabellenstruktur in der Datenblattansicht

Seit der Version Access 2003 können Sie die Tabellenstruktur auch in der Datenblattansicht ändern. Diese Möglichkeit hatte ich Ihnen bereits eingangs in Abschnitt 2.1.2, »Eine Tabelle in Access erstellen«, gezeigt (Abbildung 2.16).

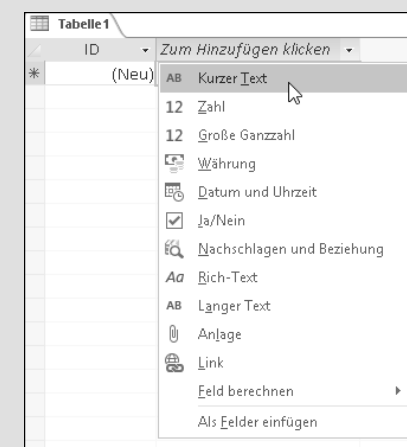


Abbildung 2.16 »Zum Hinzufügen klicken« – über dieses Menü erweitern Sie die Tabellenstruktur in der Datenblattansicht.

Allerdings offenbart das Kontextmenü nicht die ganze Wahrheit. Den vollen Zugriff auf alle Optionen finden Sie nur in der Entwurfsansicht. Genau deshalb empfehle ich Ihnen, die Tabellenstruktur *ausschließlich* in der Entwurfsansicht zu ändern.

Für die aktuelle Datenbank können Sie die Funktion zum Verändern der Tabellenstruktur in der Datenblattansicht deaktivieren. Gehen Sie dazu auf DATEI • OPTIONEN, und entfernen Sie dann im Fenster ACCESS-OPTIONEN den Haken unter AKTUELLE DATENBANK • ANWENDUNGSOPTIONEN • ENTWURFSÄNDERUNGEN FÜR TABELLEN IN DER DATENBLATTANSICHT AKTIVIEREN. Diese Änderung wirkt sich allerdings erst aus, nachdem Sie Ihre Datenbank geschlossen und wieder geöffnet haben.

Es ist jetzt an der Zeit, dass wir uns die einzelnen Felddatentypen ansehen. Anschließend zeige ich Ihnen weitere Regeln und Einschränkungen, die Sie ebenfalls in der Entwurfsansicht festlegen können.

2.3.2 Felddatentyp »Kurzer Text«

Verwenden Sie den Felddatentyp KURZER TEXT, um jegliche Art von *Zeichenfolgen* (englisch *strings*) zu speichern. Im Registerblatt ALLGEMEIN unter FELDGRÖSSE können (und sollten!) Sie die maximal zulässige Anzahl von Zeichen für ein Feld festlegen. Wenn Sie eine große Anzahl von Zeichen zulassen möchten, denken Sie bitte daran, dass die Textfelder in Formularen und Berichten auch breit genug sein müssen. Ein Feld mit dem Felddatentyp KURZER TEXT darf maximal 255 Zeichen enthalten.

Felddatentyp	englische Bezeichnung	Jet-SQL-Datentyp	Wertebereich	Speicherbedarf
KURZER TEXT	<i>Short Text</i>	CHAR VARCHAR	maximal 255 Zeichen	<ul style="list-style-type: none"> ▶ 2 Bytes je Zeichen ▶ mit aktivierter Unicode-Kompression auch weniger

Tabelle 2.6 Der Felddatentyp »Kurzer Text« eignet sich für Zeichenfolgen bis zu 255 Zeichen.

Unicode-Zeichen

Alle Zeichenfolgen speichert Access in *Unicode* ab. Sie können daher mehr als nur die 127 ASCII-Zeichen verwenden. Wichtig wird Unicode, wenn Sie eine andere Schrift statt der lateinischen, wie beispielsweise Kyrillisch, Hebräisch oder Schriften aus Fernost, verwenden möchten oder für spezielle Sonderzeichen. All dies unterstützt Access; Sie können also auch exotische Zeichen und Wörter in einer Access-Datenbank speichern.

Ein anderes Thema ist, ob alle Zeichen am Bildschirm richtig angezeigt werden. Nur die wenigsten Schriftarten unterstützen sämtliche Unicode-Zeichen. Im Office-Paket enthalten ist die Schriftart »Arial Unicode MS« mit knapp 40.000 Zeichen (und einer Größe von fast

22 MB); reicht Ihnen dieser Umfang fürs Erste? Die Anzeigeschriftart können Sie ändern, indem Sie die Tabelle in der Datenblattansicht öffnen und dann unter START • TEXTFORMATIERUNG die gewünschte Schriftart auswählen.

ID	FeldKurzerText
1	Video bietet eine leistungsstarke Möglich
2	Wenn Sie auf "Onlinevideo" klicken, könn
3	Кириллица
4	עברית
5	日本の
6	Jeg købte hot wings med pommes frites o
*	(Neu)

Abbildung 2.17 Exotische Zeichen aus dem Unicode-Bereich sind für Access kein Problem.

2.3.3 Felddatentyp »Langer Text«

Wenn Ihnen 255 Zeichen zu wenig sind, können Sie auf den Felddatentyp LANGER TEXT ausweichen. Ich sage »ausweichen« deshalb, weil Access nicht alle Funktionen für den Felddatentyp LANGER TEXT unterstützt. An diese Einschränkungen sollten Sie sich immer erinnern, wenn Sie den Felddatentyp LANGER TEXT verwenden:

- ▶ Die maximal zulässige Anzahl von Zeichen können Sie nicht festlegen.
- ▶ Sortieren in der Datenblattansicht ist gar nicht möglich.
- ▶ Beim Sortieren per Abfrage werden nur die ersten 255 Zeichen berücksichtigt.
- ▶ Beim Filtern werden nur die ersten 255 Zeichen berücksichtigt.

Diese Einschränkungen können zu bösen Überraschungen führen. Dafür können Sie aber wirklich jede Menge Zeichen speichern: bis zu 65.535 Zeichen über die Benutzeroberfläche von Access, per Programmierung sogar bis zu 2 GB!

Felddatentyp	englische Bezeichnung	Jet-SQL-Datentyp	Wertebereich	Speicherbedarf
LANGER TEXT	<i>Long Text</i>	TEXT	maximal 2 GB (über die Benutzer- oberfläche von Access: maximal 65.535 Zei- chen)	<ul style="list-style-type: none"> ▶ 2 Bytes je Zeichen ▶ mit aktivierter Unicode-Kompression auch weniger ▶ Unicode-Kompression greift nur, wenn der Inhalt größer als 4 kB ist.

Tabelle 2.7 Platz für Zeichen ohne Ende: der Felddatentyp »Langer Text«

»Langer Text« und »Memo«

Vor der Access Version 2013 hieß dieser Felddatentyp *Memo*, während *Kurzer Text* einfach nur als *Text* bezeichnet wurde. Ich finde es nur zu ärgerlich (und für den Neueinsteiger leider unnötig schwierig), dass die Bezeichnungen in der Sprache SQL wieder anders sind, nämlich CHAR für kurzen Text sowie TEXT für langen Text.

Der Felddatentyp LANGER TEXT bietet noch zwei Extras, die Sie im Registerblatt ALLGEMEIN aktivieren können:

1. **Formatierter Text:** Felder mit dem Felddatentyp KURZER TEXT werden immer ohne Formatierungen abgespeichert. Üblicherweise ist das auch für Felder mit dem Felddatentyp LANGER TEXT so.

Wenn Sie das TEXTFORMAT auf RICH-TEXT umstellen, können Sie Formatierungen einsetzen. Access speichert die Formatierungen automatisch als HTML-Code ab – nicht jedoch im Rich Text Format (RTF), wie der Name vermuten lässt.

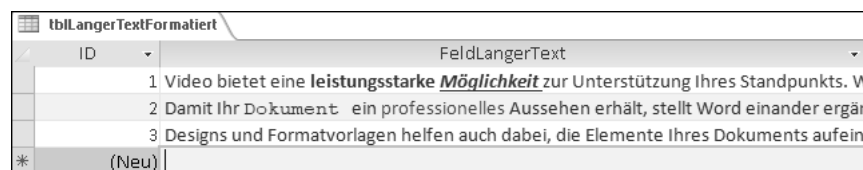


Abbildung 2.18 In einem Feld mit dem Felddatentyp »Langer Text« können Sie Formatierungen aktivieren.

2. **Protokollierung:** Wenn Sie die Option NUR ANFÜGEN auf JA setzen, protokolliert Access alle Änderungen an einem Feld mit dem Felddatentyp LANGER TEXT. Sie können sich die alten Versionen anzeigen lassen, indem Sie in der Datenblattansicht mit der rechten Maustaste auf das Feld klicken und anschließend im Kontextmenü SPALTENVERLAUF ANZEIGEN ... auswählen.



Abbildung 2.19 Nach aktivierter Protokollierung können Sie die alten Versionen des Feldinhalts aufrufen.

Protokolliert werden allerdings nur die Änderungen an dem einen Feld, nicht jedoch die unterschiedlichen Versionen des gesamten Datensatzes (für diesen Zweck können Sie die *Änderungentabelle* verwenden, die ich Ihnen in Abschnitt 2.7.6, »Typische Muster von Tabellenbeziehungen«, vorstellen werde).

2.3.4 Felddatentyp »Zahl«

Zum Speichern von Zahlen bietet Access eine ganze Reihe von Felddatentypen an, die alle unter dem Felddatentyp ZAHL zusammengefasst sind. Den eigentlichen Felddatentyp können Sie im Registerblatt ALLGEMEIN unter FELDGRÖSSE wählen. Zahlen werden grundsätzlich binär abgespeichert.

Binäre Speicherung von Zahlen

Normalerweise brauchen Sie sich keine Gedanken darüber zu machen, wie Access die Zahlen binär ablegt. Für die Ganzzahlen habe ich Ihnen trotzdem einmal grafisch dargestellt, wie eine Zahl intern in binärer Form gespeichert wird. Diese Informationen benötigen Sie, wenn Sie mit VBA-Programmierung auf die einzelnen Bits zugreifen möchten. Ein Beispielprogramm finden Sie in den Materialien zum Buch unter *02_Access_als_Datenbank_Tabellen\2.3.4_BinaerManipulation.accdb*.

1. Zu der Gruppe der *Ganzzahlen* (englisch *integer number*) gehören die Felddatentypen BYTE, INTEGER und LONG INTEGER. Der Felddatentyp BYTE kann nur positive Ganzzahlen aufnehmen und umfasst – wie der Name schon sagt – den Speicherbereich von genau einem Byte.

Bit: 7 6 5 4 3 2 1 0

Beispiele:

1 0 0 1 0 0 0 1

$$= 1 * 2^7 + 0 * 2^6 + 0 * 2^5 + 1 * 2^4 + 0 * 2^3 + 0 * 2^2 + 0 * 2^1 + 1 * 2^0$$

$$= 145$$

0 0 1 1 1 0 0 0

$$= 0 * 2^7 + 0 * 2^6 + 1 * 2^5 + 1 * 2^4 + 1 * 2^3 + 0 * 2^2 + 0 * 2^1 + 0 * 2^0$$

$$= 56$$

Abbildung 2.20 Ganzzahlen des Felddatentyps »Byte« werden in einem Byte (mit 8 Bits) ohne Vorzeichen gespeichert.

INTEGER und LONG INTEGER umfassen zwei bzw. vier Byte. Beide dieser Felddatentypen nehmen Zahlen mit Vorzeichen auf. Das Vorzeichen wird immer im höchsten Bit gespeichert.

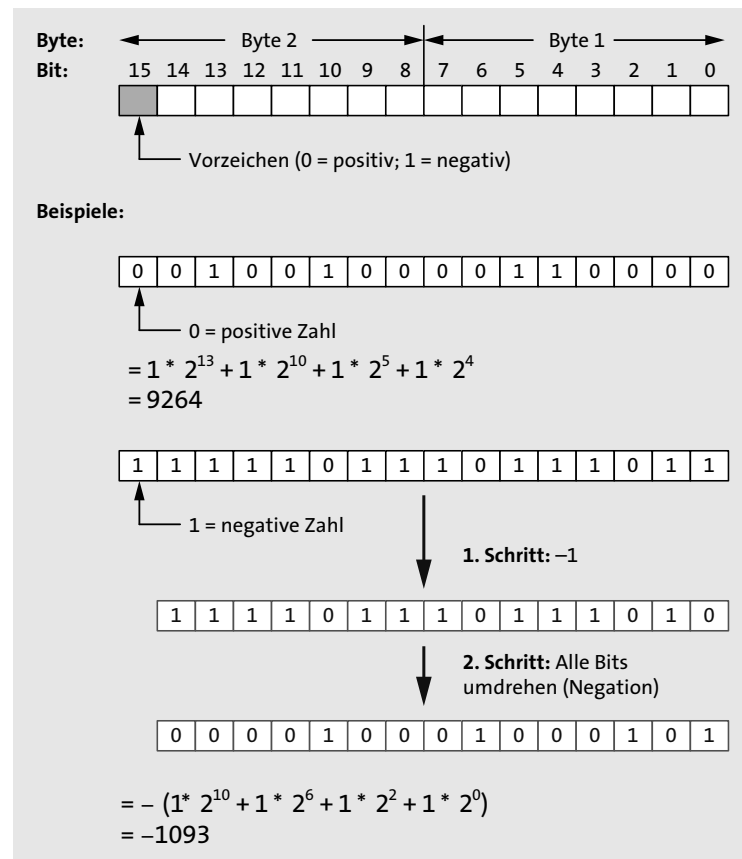


Abbildung 2.21 Ganzzahlen des Felddatentyps »Integer« werden in zwei Bytes mit Vorzeichen gespeichert. Um eine negative Zahl von der Binärdarstellung in die Dezimaldarstellung umzurechnen, sind zwei Schritte notwendig (negative Zahlen werden in der sogenannten Two's-Complement-Repräsentation gespeichert). Ganzzahlen des Felddatentyps »Long Integer« werden in vier Bytes gespeichert, alles andere ist identisch mit dem Felddatentyp »Integer«.

Mit dem Felddatentyp REPLIKATIONS-ID können Sie GUIDs speichern. Eine GUID ist eine 16 Byte große Ganzzahl, die in einem besonderen Format angezeigt wird.

Globally Unique Identifier (GUID)

GUIDs sind zunächst nichts anderes als positive Ganzzahlen, die sehr groß werden dürfen (Wertebereich: 0 ... 340.282.366.920.938.463.463.374.607.431.768.211.455). Praktischerweise werden GUIDs als Hexadezimalzahlen (siehe nächster Kasten) dargestellt.

Beispiele:

```
{9D5759B3-0EBF-4957-837D-09B1E330E998}
{C57DC5D5-B647-4966-87A6-C585264AC85C}
{C5C64BD8-A75C-48DA-98D2-FAA00F13A962}
```

Durch den großen Wertebereich ist es möglich, dass (nach festgelegten Algorithmen) generierte GUIDs praktisch weltweit eindeutig sind. Auch in Access können Sie GUIDs automatisch generieren lassen (mehr dazu in Abschnitt 2.3.5, »Felddatentyp »AutoWert««). In einer Datenbank sind GUIDs als Primärschlüssel interessant, wenn Datensätze an getrennten Standorten ohne Netzwerkverbindung erfasst und später zusammengetragen werden sollen. Solche *Replikationsszenarien* gehören zu den anspruchsvollsten Aufgaben der Datenbankprogrammierung.

Hexadezimalsystem

Das Hexadezimalsystem ist das Zahlensystem mit der Basis 16 und den 16 Ziffern 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. Zur Unterscheidung von Zahlen im Dezimalsystem wird als Präfix &H vorangestellt (gelegentlich werden auch andere Präfixe verwendet, z. B. 0x).

Beispiele:

```
&HC94B = 12 * 16^3 + 9 * 16^2 + 4 * 16^1 + 11 * 16^0 = 51531
&H16 = 1 * 16^1 + 6 * 16^0 = 22
&H0 = 0 * 16^0 = 0
```

Die Hexadezimaldarstellung ist in der Informatik besonders verbreitet, da zwei Stellen genau einem Byte (8 Bit) entsprechen.

2. *Gleitkommazahlen* (englisch *floating-point numbers*) sind Zahlen mit Nachkommastellen und Exponent. In Access können Sie sich entweder für den Felddatentyp SINGLE oder für DOUBLE entscheiden. Beide Felddatentypen unterscheiden sich in ihrer Genauigkeit, d. h. der Anzahl der Ziffern, die gespeichert werden.

Es gibt bei Gleitkommazahlen aber einen großen Haken: Nicht jede Zahl kann im Binärsystem exakt dargestellt werden. Selbst eine Zahl wie z. B. 0,1 (im Dezimalsystem exakt darstellbar) entspricht im Binärsystem dem Wert 0,0001100110011... Diese Zahl ist mit begrenzt vielen Ziffern nicht exakt zu speichern, denn sie hat eine nicht abbrechende Periode hinter dem Komma. Beim Umgang mit Gleitkommazahlen kann es also zu *Artefakten* und dadurch zu Überraschungen kommen!

Artefakte beim Umgang mit der Gleitkommaarithmetik

Gleitkommazahlen und Gleitkommaarithmetik sind ein eigenes Thema für sich, das über den Umfang dieses Buches hinausgeht. Die folgenden Artefakte können Ihnen beispielsweise beim Umgang mit Gleitkommazahlen begegnen:

- ▶ $0,1 - 1/10 = 5.5E-18$ (statt 0)
- ▶ $0,0001234567890123 + 1 = 1,0001234567890100$ (statt 1,0001234567890123)
- ▶ $1E20 + 1 = 1E20$ (statt 100000000000000000001)

Der erste Fall kommt dadurch zustande, dass unterschiedliche Genauigkeiten (SINGLE und DOUBLE) gemischt werden. Wenn in einer Rechnung gleichzeitig sehr große und sehr kleine Zahlen auftreten, kommt es zu den letzten beiden Artefakten. Diese Artefakte treten übrigens auch in Excel auf.

Am besten eignen sich Gleitkommazahlen für Messwerte oder wissenschaftliche Zahlen, die mit einem Messfehler behaftet sind. Sie können Gleitkommazahlen auch für andere Zwecke nutzen, beispielsweise um einen Prozentwert (30 % = 0,30) in einer Tabelle abzuspeichern. Nur dürfen Sie nie vergessen, dass es bei nachfolgenden Berechnungen zu Artefakten kommen kann. Achten Sie besonders darauf, Rechenergebnisse nicht auf exakte Gleichheit zu überprüfen. Noch einmal im Klartext: Verwenden Sie nicht das Gleichheitszeichen (=)!

In manchen Fällen sind Artefakte wirklich unverzeihlich, insbesondere dann, wenn es um das liebe Geld geht. Verwenden Sie für Geldbeträge *niemals* den Felddatentyp SINGLE oder DOUBLE! Für Geldbeträge gibt es in Access den Datentyp WÄHRUNG, den ich Ihnen in Kürze vorstellen werde; und nur dort gehören Geldbeträge hinein.

3. Neben den Gleitkommazahlen gibt es in Access einen Datentyp für *exakte Dezimalzahlen* (Festkommazahlen, englisch *fixed-point numbers*), den Felddatentyp DEZIMAL. Im Registerblatt ALLGEMEIN können Sie unter GENAUIGKEIT die Anzahl der Stellen angeben (maximal 28). Unter DEZIMALSTELLEN legen Sie fest, wie viele der Stellen hinter dem Komma stehen (maximal der Wert von GENAUIGKEIT). Im Gegensatz zu Gleitkommazahlen bleiben die Stellen hinter dem Komma exakt so, wie Sie es festgelegt haben, auch bei Berechnungen. Sie können unter DEZIMALSTELLEN auch den Wert 0 eintragen und auf diese Weise sehr große Ganzzahlen speichern (größere als mit dem Felddatentyp LONG INTEGER).

Der Felddatentyp DEZIMAL wird vor allem in Server-Datenbanken verwendet (beispielsweise Microsoft SQL Server oder Oracle Database; dort aber unter einem anderen Namen als in Access, nämlich »Decimal« bzw. »Number«). Der DEZIMAL-Felddatentyp der Server-Datenbanken kann noch einiges mehr, u. a. können Sie bis zu 38 Stellen speichern.

In Access wurde dieser Felddatentyp erst relativ spät mit der Version Access 2000 eingeführt. Am Anfang gab es in Access eine Reihe von Fehlern und anderen Unzulänglichkeiten. Dazu gehört, dass es bis heute keinen entsprechenden DEZIMAL-Datentyp in Visual Basic for Applications (VBA) gibt. Wegen dieser schlechten Erfahrungen wird der Felddatentyp DEZIMAL in Access immer noch selten eingesetzt. Vielleicht liegt das auch daran, dass der Wertebereich (nicht jedoch die Genauigkeit!) der Gleitkommazahlen so verlockend aussieht. Mein Eindruck ist, dass die meisten Kinderkrankheiten des Felddatentyps

DEZIMAL mittlerweile behoben sind. Für exakte Dezimalzahlen bevorzuge ich daher den Felddatentyp DEZIMAL gegenüber SINGLE oder DOUBLE.

Als Zusammenfassung hier noch einmal alle Felddatentypen, die Sie für Zahlen einsetzen können:

Felddatentyp	englische Bezeichnung	Jet-SQL-Datentyp	Wertebereich	Speicherbedarf
BYTE	<i>Byte</i>	TINYINT	0 ... 255 &H0 ... &HFF	1 Byte
INTEGER	<i>Integer</i>	SMALLINT	-32.768 ... +32.767 -&H8000 ... +&H7FFF	2 Byte
LONG INTEGER	<i>Long Integer</i>	INTEGER	-2.147.483.648 ... + 2.147.483.647 -&H80000000 ... +&H7FFFFFFF	4 Byte
REPLIKATIONS-ID	<i>Replication ID</i>	UNIQUE-IDENTIFIER	&H0 ... &HFFFFFFFFFFFFFFF FFFFFFFFFFFFFFF	16 Byte
SINGLE	<i>Single</i>	REAL	-3,402823E38 ... -1,401298E-45 0 +1,401298E-45 ... +3,402823E38	4 Byte
DOUBLE	<i>Double</i>	FLOAT	-1,79769313486232E308 ... -4,94065645841247E-324 0 +4,94065645841247E-324 ... +1,79769313486232E308	8 Byte
DEZIMAL	<i>Decimal</i>	DECIMAL	-1E-28 ... +1E+28	17 Byte

Tabelle 2.8 Bei den Felddatentypen für Zahlen sind Ganzzahlen (»Byte«, »Integer«, »Long Integer«, »Replikations-ID«), Gleitkommazahlen (»Single«, »Double«) und Festkommazahlen (»Dezimal«) zu unterscheiden. Bei den Ganzzahlen habe ich den Wertebereich auch in Hexadezimalzahlen angegeben. Sehr wichtig für die Gleitkommazahlen (»Single« und »Double«) ist die unterschiedliche Genauigkeit (7 Stellen bzw. 15 Stellen). Trotzdem kann mit Gleitkommazahlen nicht jede rationale Zahl exakt abgebildet werden. Mathematisch ausgedrückt: Es gibt Lücken im Wertebereich.

2.3.5 Felddatentyp »AutoWert«

Beim Felddatentyp `AUTOWERT` vergibt Access die Werte – wie der Name schon sagt – automatisch. Ein `AUTOWERT` eignet sich daher hervorragend für den Primärschlüssel. Unter `ALLGEMEIN • FELDDGRÖSSE` können Sie zwischen `LONG INTEGER` und `REPLIKATIONS-ID` wählen. Sie kennen diese beiden Felddatentypen bereits aus dem vorhergehenden Abschnitt, denn eigentlich ist der `AUTOWERT` nur eine besondere Form des Felddatentyps `ZAHL`. Der einzige Unterschied zum Felddatentyp `ZAHL` ist, dass Access die Werte automatisch vergibt. In ein Feld mit dem Felddatentyp `AUTOWERT` können Sie also weder einen Wert selbst eintragen noch können Sie einen von Access vergebenen Wert ändern.

1. Für `LONG INTEGER` können Sie unter `ALLGEMEIN • NEUE WERTE` auswählen, ob Access aufsteigende Werte beginnend mit der Zahl Eins (`INKREMENT`) oder zufällige Zahlen erzeugt (`ZUFALL`).

Den Zähler für »AutoWert«-Felder zurücksetzen

Übrigens setzt Access den Zähler für `INKREMENT` nicht zurück, wenn Sie Datensätze löschen. Wenn Sie alle Datensätze einer Tabelle löschen und dann wieder mit dem Wert 1 beginnen möchten, klicken Sie auf `DATENBANKTOOLS • DATENBANK KOMPRIMIEREN UND REPARIEREN`. Dadurch werden die Zähler aller Tabellen zurückgesetzt.

2. Im Fall der `REPLIKATIONS-ID` erzeugt Access GUIDs (siehe den Kasten »Globally Unique Identifier«), die mit an Sicherheit grenzender Wahrscheinlichkeit weltweit eindeutig sind.

In Abschnitt 2.7, »Beziehungen zwischen Tabellen«, werde ich Ihnen *Fremdschlüssel* vorstellen, mit denen Sie zwei Tabellen verbinden können. Ohne viel vorgreifen zu wollen, möchte ich Sie jetzt schon darauf hinweisen, dass die Feldgröße von Primär- und Fremdschlüssel identisch sein muss. Am besten entscheiden Sie sich bei allen ID-Feldern einer Datenbank entweder für `LONG INTEGER` oder für `REPLIKATIONS-ID`. Ich empfehle Ihnen `LONG INTEGER`, denn spätestens bei der Programmierung ist dies viel angenehmer als der Umgang mit den großen GUIDs (Felddatentyp `REPLIKATIONS-ID`).

Felddatentyp	englische Bezeichnung	Jet-SQL-Datentyp	Wertebereich	Speicherbedarf
<code>AUTOWERT</code> <code>LONG INTEGER</code>	<i>AutoNumber</i> <i>Long Integer</i>	COUNTER	–2.147.483.648 ... + 2.147.483.647 –&H80000000 ... +&H7FFFFFFF	4 Byte

Tabelle 2.9 Die beiden Felddatentypen der Gruppe »AutoWert« sind identisch mit den beiden Felddatentypen »Long Integer« und »Replikations-ID«. Einziger Unterschied: Access vergibt die Werte automatisch.

Felddatentyp	englische Bezeichnung	Jet-SQL-Datentyp	Wertebereich	Speicherbedarf
<code>AUTOWERT</code> <code>REPLIKATIONS-ID</code>	<i>AutoNumber</i> <i>Replication ID</i>	–	&H0 ... &HFFFFFFFFFFFFFFF FFFFFFFFFFFFFFF	16 Byte

Tabelle 2.9 Die beiden Felddatentypen der Gruppe »AutoWert« sind identisch mit den beiden Felddatentypen »Long Integer« und »Replikations-ID«. Einziger Unterschied: Access vergibt die Werte automatisch. (Forts.)

Ein `AUTOWERT`-Feld ist *nicht* automatisch der Primärschlüssel. Dazu sind noch zwei weitere Handgriffe notwendig:

1. Wählen Sie das `AUTOWERT`-Feld in der Entwurfsansicht aus.
2. Klicken Sie auf `TABELLENTOOLS • ENTWURF • TOOLS • PRIMÄRSCHLÜSSEL`.

Feldname	Felddatentyp
FlugID	AutoWert
AbflugDatum	Kurzer Text
AbflugZeit	Kurzer Text
AnkunftZeit	Kurzer Text
Fluggesellschaft	Kurzer Text
Flugnummer	Kurzer Text
AbflugFlughafen	Kurzer Text
AnkunftFlughafen	Kurzer Text
Passagiere	Kurzer Text

Abbildung 2.22 Der Primärschlüssel wird durch einen kleinen Schlüssel vor dem Feldnamen gekennzeichnet.

2.3.6 Felddatentyp »Datum/Uhrzeit«

Für alles, was irgendwie mit Zeitangaben zu tun hat, ist der Felddatentyp `DATUM/UHRZEIT` vorgesehen. Intern speichert Access einen Zeitpunkt als Zahl ab, nämlich als Anzahl der Tage, die seit dem 30.12.1899 um 0:00 Uhr vergangen sind. Wie geht dann aus der Zahl die Uhrzeit hervor? Genauer gesagt speichert Access ein Datum als Gleitkommazahl vom Felddatentyp `DOUBLE` ab. Die Stellen vor dem Komma ergeben den Tag, während mit den Nachkommastellen die Zeit berechnet werden kann. Dazu ein paar Beispiele in Tabelle 2.10.

In der Praxis werden Sie nicht den internen `DOUBLE`-Wert, sondern nur die Anzeige als Datum und Uhrzeit zu Gesicht bekommen. In Abschnitt 2.6, »Formatierungen in Tabellen«, werde ich Ihnen ausführlich zeigen, wie Sie das Anzeigeformat anpassen können.

»Double«-Wert	Tage seit dem 30.12.1899	Datum	Zeitangabe	Uhrzeit
1,0	1	31.12.1899	,0	00:00:00
22194,5	22194	05.10.1960	,5	12:00:00
41180,6875	41180	28.09.2012	,6875	16:30:00
-54655,2988425926	-54655	10.05.1750	,2988425926	07:10:20

Tabelle 2.10 Intern speichert Access Zeitangaben als Zahlen des Felddatentyps »Double« ab. Die Festlegung auf das Bezugsdatum 30.12.1899 stammt noch aus den Urzeiten von Microsoft Excel und Lotus 1-2-3.

Da Sie jetzt wissen, dass Access ein Datum mit Uhrzeit intern als Zahlenwert speichert, können Sie dies ausnutzen:

1. Sortieren nach Datum
2. Vergleichen von Zeitangaben
3. Filtern von Zeitangaben

Epoche, Wertebereich und Genauigkeit von Zeitangaben

Es ist in der Informatik weitverbreitet, Zeitangaben als Zahlenwerte zu speichern. Die Form ist jedoch nicht einheitlich und unterscheidet sich von Programm zu Programm. Mit Form meine ich hier zum einen den Zeitpunkt 0, den man *Epoche* nennt. Zum anderen sind das der Wertebereich und die Auflösung. Für Access gilt Folgendes für die Form:

- ▶ Epoche ist der 30.12.1899 um 0:00 Uhr.
- ▶ Das kleinste zulässige Datum ist der 01.01.0100 (das Jahr 100 n. Chr.).
- ▶ Das größte zulässige Datum ist der 31.12.9999 (das Jahr 9999 n. Chr.).
- ▶ Die Auflösung beträgt 1 s, das heißt, Zeitangaben werden auf die Sekunde genau gespeichert.

Die Auflösung könnte sogar noch höher sein, denn der DOUBLE-Wert unterstützt 15 Stellen. Mit den üblichen Bordmitteln von Access können Sie Millisekunden oder noch kleinere Sekundenbruchteile jedoch nicht erfassen oder bearbeiten. In sehr vielen Fällen reicht die Genauigkeit von einer Sekunde aber aus.

Vergessen Sie bitte nicht, dass Epoche, Wertebereich und Genauigkeit in anderen Programmen unterschiedlich sein können. Beispielsweise speichert der Microsoft SQL Server Zeitangaben auf die Millisekunde genau, mit einem anderen Wertebereich und mit einer anderen Epoche!

Der Felddatentyp DATUM/UHRZEIT geht korrekt mit Schaltjahren um. Wenn Sie versuchen, den 29.02.2019 einzugeben, erhalten Sie eine Fehlermeldung.

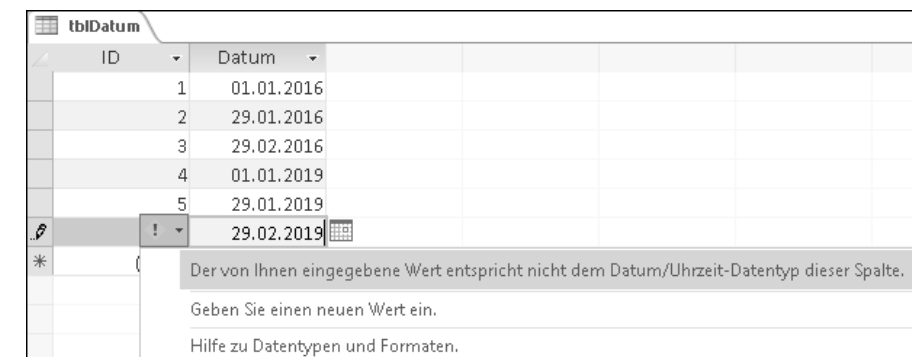


Abbildung 2.23 Die Fehlermeldung ist vielleicht etwas unverständlich, aber richtig: Der 29.02.2019 ist ein ungültiges Datum, denn das Jahr 2019 ist kein Schaltjahr.

Bei der Zeitrechnung gibt es aber noch ein paar fiese Gemeinheiten, die Access nicht beherrscht. Für den Felddatentyp DATUM/UHRZEIT unbekannt sind:

1. Schaltsekunden

Bisher gab es 37 Schaltsekunden, um Unregelmäßigkeiten und die Verlangsamung der Erdrotation auszugleichen.

2. Andere Kalendertypen

Verschiedene Kalendertypen sind insofern relevant, als der heute weltweit verwendete Kalender (der *Gregorianische Kalender*) erst im Jahr 1582 eingeführt wurde. Davor wurde im Mittelalter der *Julianische Kalender* verwendet, der ein paar Probleme mit den Schaltjahren hatte. Bis 1582 war der Julianische Kalender schon etwas aus dem Ruder gelaufen, so dass bei der Umstellung einfach 10 Tage übersprungen wurden. Die Tage vom 05.10.1582 bis einschließlich 14.10.1582 gab es also nie ...

Diese Feinheiten sind Access jedoch unbekannt. Es ist daher wenig sinnvoll, Kalenderangaben vor dem 15.10.1582 exakt anzugeben. In anderen Ländern fand die Umstellung auf den Gregorianischen Kalender noch später statt (z. B. im Britischen Weltreich einschließlich der amerikanischen Kolonien erst im Jahr 1752). Die exakten Zeitangaben in der frühen Neuzeit sind also durchaus problematisch.

3. Zeitzonen

Angaben zur Zeitzone sollten Sie in einem zusätzlichen Feld abspeichern, wenn dies notwendig ist. In diesem Fall empfehle ich Ihnen, Zeitangaben grundsätzlich in der *Universalzeit* (UTC) abzuspeichern (mehr dazu im Kasten »Coordinated Universal Time (UTC) – die Universalzeit«).

4. Winterzeit und Sommerzeit

Üblicherweise wird bei der Angabe der Zeitzone vermerkt, ob die Winterzeit (z. B. *mitteleuropäische Zeit, MEZ*; englisch *Central European Time, CET*) oder die Sommerzeit (z. B. *mitteleuropäische Sommerzeit, MESZ*; englisch *Central European Summer Time, CEST*) gilt. Sollte es also relevant sein, so empfehle ich Ihnen auch hier wärmstens die Universalzeit UTC.

Für diese Sonderfälle in der Zeitrechnung müssen sie leider selbst Sorge tragen, beispielsweise durch ein zusätzliches Feld »Zeitzone«.

Coordinated Universal Time (UTC) – die Universalzeit

Ich erinnere mich noch gut an eine Projektarbeit während meines Studiums, an der vier Studenten beteiligt waren. Jede Woche mussten wir bis freitags um 14:00 Uhr Ergebnisse bei unserem Dozenten per E-Mail einreichen. Etwas hinderlich war, dass wir in diesem Semester zu Praktika an verschiedenen Orten tätig waren: Der Erste hatte sicherlich das abenteuerlichste Praktikum von uns allen, denn er befand sich mitten in Sibirien in Irkutsk. Ein anderer Kommilitone war in Philadelphia, ich selbst in Boston, und der vierte aus dem Team war in Hannover geblieben. Die verschiedenen Zeitzonen führen da schnell zu Missverständnissen: »Meinst Du jetzt 10 Uhr deine oder meine Zeit?«

Immer dann, wenn verschiedene Zeitzonen oder die Sommerzeit stört, hilft die universelle Zeit UTC. Die universelle Zeit läuft nämlich immer normal weiter, sie kennt keine Sprünge wie z. B. unterschiedliche Zeitzonen. Wenn es in Deutschland 10 Uhr UTC ist, dann ist das an jedem anderen Ort in der Welt ebenso. Bezogen auf meine Studienarbeit: Mit UTC hätten wir es einfach gehabt, denn ein Treffen um 10 Uhr UTC lässt sich einfacher planen.

UTC entspricht der Ortszeit von Großbritannien und anderen Ländern der Zeitzone 0 – allerdings nur in der Winterzeit. Für Deutschland gilt eine Stunde mehr, daher während der Winterzeit UTC + 1 h bzw. während der Sommerzeit UTC + 2 h. Für andere Zeitzonen gibt es entsprechende Angaben (z. B. UTC–5 für die Ostküste der USA, Winterzeit).

Wenn Sie Zeitpunkte in verschiedenen Zeitzonen erfassen möchten, sollten Sie alle Zeitangaben als UTC ablegen. In einem zusätzlichen Feld können Sie die Zeitzone (*UTC-Offset*) angeben. Nur so können Sie die Zeitpunkte richtig miteinander vergleichen. Hier ein paar Beispiele:

- ▶ Termine, die über verschiedene Zeitzonen hinweg koordiniert werden sollen
- ▶ Flüge über verschiedene Zeitzonen hinweg
- ▶ Laufzeiten und Messwerte einer Windturbine, die auch während der Umstellung von Winterzeit auf Sommerzeit bzw. umgekehrt in Betrieb ist

Übrigens läuft die interne Uhr Ihres Computers auch mit UTC. Die universelle Zeit UTC ist also nichts Exotisches, sondern etwas sehr Praktisches.

Felddatentyp	englische Bezeichnung	Jet-SQL-Datentyp	Wertebereich	Speicherbedarf
DATUM/UHRZEIT	<i>Date/Time</i>	DATETIME	01.01.0100 ... 31.12.9999	8 Byte

Tabelle 2.11 Zeitangaben speichert Access auf die Sekunde genau ab.

2.3.7 Felddatentyp »Währung«

Eine der wichtigsten Sachen der Welt ist bekanntlich das liebe Geld. Dafür gibt es in Access einen eigenen Felddatentyp, nämlich WÄHRUNG. Dies ist eine Festkommazahl mit 15 Stellen vor dem Komma und vier Nachkommastellen. Im Gegensatz zu Gleitkommazahlen können Sie hiermit exakte Berechnungen durchführen, was bei Geldbeträgen besonders wichtig ist.

Auch wenn ich es schon mehrfach erwähnt habe, möchte ich an dieser Stelle noch einmal daran erinnern: Verwenden Sie *niemals* Gleitkommazahlen (Felddatentyp SINGLE oder DOUBLE), um Geldbeträge zu speichern! Die Artefakte der Gleitkommaarithmetik sind für ein so sensibles Thema wie Geld einfach nicht akzeptabel. Denken Sie auch daran, dass Ihre Kollegen einer Datenbank, die Sie erstellt haben, vertrauen und diese als Werkzeug akzeptieren sollen. Ein Killerargument wie »Ihre Datenbank kann ja nicht einmal richtig rechnen!« wäre doch eine wahre Katastrophe, oder?

Mit dem Felddatentyp WÄHRUNG speichert Access nicht das Währungssymbol. Wenn Sie nur mit einer Währung arbeiten, ist dies nicht weiter schlimm. Angezeigt wird das standardmäßige Währungssymbol Ihrer Windows-Einstellungen. Wenn gewünscht, können Sie über Formatierung ein anderes Währungssymbol festlegen (siehe Abschnitt 2.6, »Formatierungen in Tabellen«).

Wenn Sie in einer Tabelle Geldbeträge in unterschiedlichen Währungen ablegen wollen, gibt es nur einen sauberen Weg: ein separates Feld, in dem die Währung gespeichert wird. Dafür eignet sich beispielsweise der Währungscode nach dem ISO-Standard 4217, der immer aus drei Buchstaben besteht:

Währungscode	Währung
EUR	Euro
GBP	Britische Pfund Sterling
USD	US-Dollar
JPY	Japanische Yen

Tabelle 2.12 Einige Währungscode nach dem ISO-Standard 4217

Felddatentyp	englische Bezeichnung	Jet-SQL-Datentyp	Wertebereich	Speicherbedarf
WÄHRUNG	Currency	MONEY	-922.337.203.685.477,5808 ... 922.337.203.685.477,5807	8 Byte

Tabelle 2.13 Der Felddatentyp »Währung« wird intern als Festkommazahl mit vier Nachkommastellen gespeichert.

Der Felddatentyp »Währung« wird manchmal auch zweckentfremdet

Bevor in Access 2000 der Felddatentyp DEZIMAL eingeführt wurde, war WÄHRUNG der einzige Felddatentyp für Festkommazahlen. Deshalb wurde und wird er gerne zweckentfremdet, nämlich für exakte Dezimalzahlen mit bis zu vier Nachkommastellen. Mit einer geeigneten Formatierung (siehe Abschnitt 2.6, »Formatierungen in Tabellen«) können Sie in solchen Fällen das Währungssymbol unterdrücken.

Ich bin eigentlich immer Befürworter davon, so nahe wie möglich an der Realität zu bleiben. Konkret würde das bedeuten, dass dort, wo Währung draufsteht, wirklich nur Geld drinsteckt. Ich finde den Felddatentyp WÄHRUNG aber derart praktisch (vor allem, was die Programmierung angeht), dass ich hier meinen Prinzipien untreu werde.

Wie auch immer Sie sich entscheiden, Sie sollten hierzu eine einheitliche Regelung zu Projektbeginn treffen. Das gilt umso mehr, wenn Sie in einem Team arbeiten. Achten Sie bitte auf aussagekräftige Feldnamen (z. B. »MehrwertsteuerProzent«), damit jeder sofort erkennt, welche Werte tatsächlich abgespeichert werden (in diesem Beispiel: Prozentsatz statt Währung).

2.3.8 Felddatentyp »Ja/Nein«

Für die einfachste Art von Information gibt es den Felddatentyp JA/NEIN. Eine andere Bezeichnung ist *boolesche Variable* (englisch *boolean*) oder *Flag*. Praktischerweise zeigt Access in der Datenblattansicht ein Ja/Nein-Feld als *Kontrollkästchen* (englisch *checkbox*) an, in der Sie einen Häkchen setzen und wieder entfernen können.

Felddatentyp	englische Bezeichnung	Jet-SQL-Datentyp	Wertebereich	Speicherbedarf
JA/NEIN	Yes/No (Boolean)	BIT	»Ja« oder »Nein«	1 Byte

Tabelle 2.14 In einem »Ja/Nein«-Feld sind nur die Werte »Ja« (»wahr«, englisch »true«) oder »Nein« (»falsch«, englisch »false«) zulässig.

2.3.9 Felddatentyp »OLE-Objekt«

Für große Binärdaten ist der Felddatentyp OLE-OBJEKT geeignet. Dieser Felddatentyp ist verwandt mit dem Felddatentyp LANGER TEXT, den ich in Abschnitt 2.3.3, »Felddatentyp »Langer Text««, besprochen habe. Auch für OLE-Objekte gelten nämlich ein paar Besonderheiten:

- ▶ Maximal 2 GB Binärdaten können pro Feld gespeichert werden.
- ▶ Die maximal zulässige Anzahl von Bytes können Sie nicht festlegen.
- ▶ Sortieren und Filtern von Binärdaten ist nicht möglich (und auch nicht sinnvoll).

Access nutzt für Binärdaten den Mechanismus der *Objektverknüpfung und -einbettung* (englisch *Object Linking and Embedding, OLE*), woraus sich auch der Name dieses Felddatentyps ableitet. Sie können eine beliebige Datei in das Feld speichern (oder verknüpfen). Access speichert zusätzlich zum Dateinhalt weitere Informationen darüber ab, mit welchem Programm die Datei später einmal wieder geöffnet werden kann.

Diesen OLE-basierten Ansatz können Sie auch verwenden, um Bilder in Ihrer Datenbank abzuspeichern. In Formularen und Berichten können die Bilder später auch dargestellt werden. Dies ist sicherlich der häufigste Einsatz des Felddatentyps OLE-OBJEKT.

Felddatentyp	englische Bezeichnung	Jet-SQL-Datentyp	Wertebereich	Speicherbedarf
OLE-OBJEKT	OLE Object	IMAGE	maximal 2 GB Daten	Größe der Binärdaten + zusätzliche OLE-Informationen

Tabelle 2.15 Platz für Binärdaten ohne Ende: der Felddatentyp »OLE-Objekt«

2.3.10 Andere Felddatentypen

In den letzten Abschnitten haben Sie alle wichtigen Felddatentypen kennengelernt. Es gibt noch ein paar Sonderlinge:

1. Der **Felddatentyp »Binär«**
Neben dem OLE-OBJEKT gibt es den Felddatentyp BINÄR zum Speichern von Binärdaten. Mit diesem Felddatentyp können Sie maximal 255 Byte speichern, was verschwindend wenig ist im Vergleich zu 2 GB für ein OLE-OBJEKT. Daher wird der Felddatentyp BINÄR heute so gut wie überhaupt nicht mehr verwendet. Sie können ihn nicht einmal in der Entwurfsansicht einer Tabelle auswählen! Wenn Sie diesen exotischen Felddatentyp unbedingt einsetzen wollen, hilft nur die Programmierung – aber es gibt ihn noch.
2. Der **Felddatentyp »Anlage«**
Mit dem Felddatentyp ANLAGE bietet Access Ihnen eine weitere Möglichkeit, Binärdaten in Ihrer Datenbank zu speichern. Im Unterschied zum OLE-OBJEKT kann ANLAGE mehr als

nur eine Datei pro Feld speichern. ANLAGE ist der modernere Weg, Bilder oder Binärdateien innerhalb einer Datenbank zu speichern.

Welcher Felddatentyp ist besser: »Anlage« oder »OLE-Objekt«?

Ich empfehle Ihnen, üblicherweise den Felddatentyp ANLAGE einzusetzen. Das ist der modernere Weg und erspart Ihnen ein paar Altlasten von OLE (Mechanismus der Objektverknüpfung und -einbettung).

Wenn Sie jedoch sicherstellen möchten, dass in einem Feld wirklich nur eine einzige Datei abgespeichert werden kann, dann müssen Sie nach wie vor den Felddatentyp OLE-OBJEKT verwenden. In meinen Augen ist dies eine wichtige Überlegung in Bezug auf das Datenbankdesign. Daneben fallen mir drei technische Einschränkungen ein, die den Felddatentyp OLE-OBJEKT unverzichtbar machen:

- ▶ Das Verlinken zu einer Datei im Dateisystem geht nur über OLE-OBJEKT.
- ▶ Server-Datenbanken (beispielsweise der Microsoft SQL Server) kennen den Felddatentyp ANLAGE nicht.
- ▶ Der Felddatentyp ANLAGE blockiert bestimmte Dateitypen (unter anderem .exe-Dateien).

3. Der Felddatentyp »Link«

Schließlich gibt es noch den Felddatentyp LINK für Hyperlinks. Das Schöne an diesem Felddatentyp ist, dass Sie einen Hyperlink direkt im Feld anklicken können und dadurch die entsprechende Webseite in Ihrem Standardbrowser geöffnet wird.

Felddatentyp	englische Bezeichnung	Jet-SQL-Datentyp	Wertebereich	Speicherbedarf
BINÄR	<i>Binary</i>	BINARY	maximal 255 Byte Daten	Anzahl der Byte
ANLAGE	<i>Attachment</i>	–	maximal 256 MB pro Datei	Größe der Binärdaten
LINK	<i>Hyperlink</i>	–	maximal 2.048 Zeichen	2 Bytes je Zeichen

Tabelle 2.16 Mit dem Felddatentyp »Anlage« können Sie auch mehrere Dateien in einem Feld abspeichern.

2.3.11 Berechnete Felder

Eigentlich sind Tabellen nur zum Erfassen, Ändern und Löschen von Daten da. Die Auswertung von Daten – und dazu zählen auch Berechnungen – hat hier üblicherweise nichts verloren, sondern gehört zum Thema *Abfragen*. In Abschnitt 3.2, »Auswerten von Daten eines Da-

tensatzes: Berechnete Felder«, werde ich Ihnen zeigen, wie Sie Berechnungen mit Hilfe von Abfragen durchführen können.

Sehr zu meinem Leidwesen gibt es in Access inzwischen den Felddatentyp BERECHNET. Ich mag diesen Felddatentyp deswegen nicht, weil er den Prozess des Datenbankdesigns nur unnötig verkompliziert. Sie werden sich die Arbeit deutlich einfacher machen, wenn Sie Berechnungen dort lassen, wo Sie hingehören: und zwar bei den Abfragen.

Der Vollständigkeit halber zeige ich Ihnen trotzdem, wie Sie innerhalb einer Tabelle berechnete Felder verwenden können. Wenn Sie den Felddatentyp BERECHNET auswählen, zeigt Ihnen Access automatisch das Fenster AUSDRUCKS-GENERATOR an:

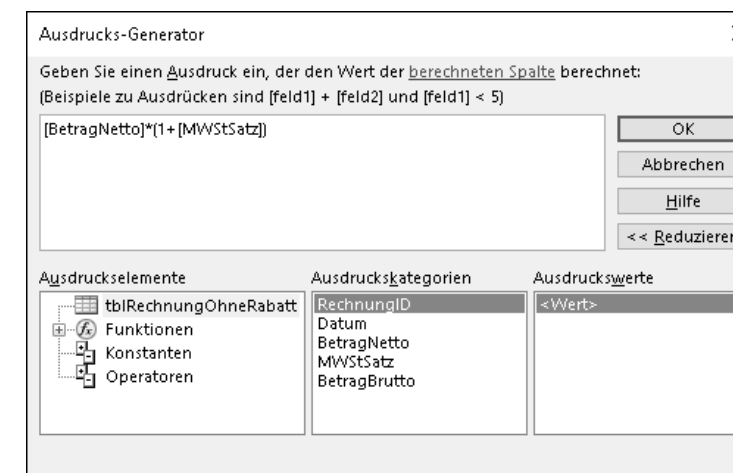


Abbildung 2.24 Mit dem Ausdrucks-Generator können Sie eine Formel zur Berechnung eines berechneten Feldes erstellen.

Tragen Sie hier die *Formel* zur Berechnung ein. Sinnvollerweise greift eine Berechnung Werte aus anderen Feldern auf. Schreiben Sie dazu die Feldnamen in eckigen Klammern ([und]). Hier ein paar Beispiele:

1. Berechnung des Gesamtbetrags mit Mehrwertsteuer:

$$[\text{BetragNetto}] * (1 + \text{MWStSatz})$$

2. Auf einen Betrag einen bestimmten Rabatt gewähren:

$$[\text{BetragNetto}] * (1 - [\text{Rabatt}])$$

3. Erst einen bestimmten Rabatt gewähren, dann die Mehrwertsteuer berechnen

$$[\text{BetragNetto}] * (1 - [\text{Rabatt}]) * (1 + \text{MWStSatz})$$

Nachdem Sie auf OK geklickt und den Ausdrucks-Generator dadurch geschlossen haben, finden Sie Ihre Formel zur Berechnung im Registerblatt ALLGEMEIN unter AUSDRUCK. Unter ERGEBNISTYP können Sie abschließend den gewünschten Felddatentyp für das Ergebnis der Berechnung auswählen. Ich habe diese Beispiele und ein paar weitere Berechnungen in die

Datenbank *O2_Access_als_Datenbank_Tabellen\2.3.11_Berechnete_Felder.accdb* aus den Materialien zum Buch gepackt.

Kompatibilität mit SharePoint

Mein Eindruck ist, dass Microsoft innerhalb der letzten Jahre Access mit viel Aufwand in Richtung SharePoint ausgerichtet hat. Und genau hierfür, nämlich für die Kompatibilität mit Microsoft SharePoint, ist der Felddatentyp `BERECHNET` notwendig.

2.3.12 Feldeinschränkungen

Sie kennen jetzt alle Felddatentypen von Access. Wenn Sie für ein Feld den passenden Felddatentyp auswählen, kann Access damit für konsistente Daten sorgen (und beispielsweise verhindern, dass in einem Zahlenfeld Text abgespeichert wird).

Neben den Felddatentypen können und sollten Sie für ein Feld weitere Einschränkungen festlegen. Access wird auch diese *Feldeinschränkungen* überprüfen, bevor ein neuer Wert abgespeichert wird. Wir werden uns in diesem Abschnitt mit den folgenden Feldeinschränkungen beschäftigen:

1. EINGABE ERFORDERLICH
2. Feld-Gültigkeitsregel
3. STANDARDWERT

Bevor wir uns die einzelnen Feldeinschränkungen ansehen, möchte ich an Sie appellieren, hiervon auch rigoros Gebrauch zu machen. Nur so kann Access überhaupt erst Dateneingaben vor dem Abspeichern prüfen und auf diesem Weg die Datenkonsistenz gewährleisten.

Eingabe erforderlich

Eine sehr wichtige Feldeinschränkung ist `EINGABE ERFORDERLICH`. Beispielsweise ist es wenig sinnvoll, einen Flug ohne Datum, Abflugszeit, Zeit der Landung, Angaben zu den Flughäfen und ohne Flugnummer anzugeben. Alle diese Felder sollten gefüllt sein.

1. Öffnen Sie die Datenbank *O2_Access_als_Datenbank_Tabellen\2.3.12_Fluege.accdb* aus den Materialien zum Buch.
2. Öffnen Sie die Tabelle *tblFlug* in der Entwurfsansicht.
3. Wählen Sie das Feld »AbflugDatum« aus.
4. Wählen Sie im Registerblatt ALLGEMEIN unter `EINGABE ERFORDERLICH` den Wert JA aus.
5. Setzen Sie auch für alle anderen Felder mit Ausnahme von »FlugID« und »Passagiere« `EINGABE ERFORDERLICH` auf JA.

Das Ergebnis finden Sie in den Materialien zum Buch unter *O2_Access_als_Datenbank_Tabellen\2.3.12_Fluege_Eingabe_erforderlich.accdb*.

Ein `AUTOWERT`-Feld wird von Access automatisch eingetragen, daher können Sie für »FlugID« die Feldeinschränkung `EINGABE ERFORDERLICH` nicht verändern. Jetzt ist nur noch das Feld »Passagiere« optional und darf leer bleiben. Für alle anderen Felder wird Access vor dem Abspeichern des Datensatzes überprüfen, ob wirklich ein Wert eingetragen ist.

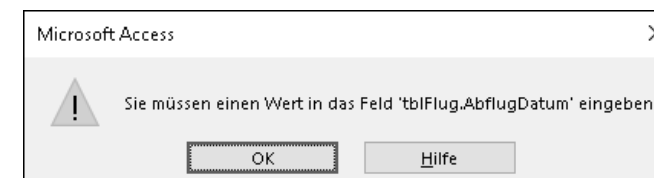


Abbildung 2.25 Eine solche Fehlermeldung erscheint, wenn Sie vergessen haben, ein Feld zu füllen, für das die Eingabe erforderlich ist.

Den Fall »ein Feld bleibt leer« sollten wir uns noch genauer ansehen. In der Welt der Datenbanken hat ein leeres Feld den Inhalt `NULL`. Am besten passt »nicht definiert« als Bedeutung für `NULL`. Wenn Sie einen neuen Datensatz erstellen, sind zunächst alle Felder `NULL` (also nicht definiert; `NULL` ist nämlich streng genommen kein Wert). Die Feldeinschränkung `EINGABE ERFORDERLICH` bezeichnet man auch als *NOT NULL* (Abkürzung: *NN*). Mit anderen Worten: Beim Abspeichern auf keinen Fall `NULL` akzeptieren.

NULL und die Zahl Null (numerisch 0)

Auch wenn es so ähnlich klingt, hat `NULL` überhaupt nichts mit der Zahl 0 gemeinsam. Wenn ich 0 Euro auf meinem Konto habe, sind meine Ersparnisse aufgebraucht. `NULL` bedeutet hingegen, dass ich meinen Kontostand gerade nicht kenne.

Access und VBA verwenden die Schreibweise `Null`, was ich für Formeln und Listings übernommen habe. Um Missverständnisse zu vermeiden, verwende ich im Text die Schreibweise `NULL` in Großbuchstaben.

Felder, für die `EINGABE ERFORDERLICH` auf JA gesetzt ist, werden im Fachjargon als *NOT-NULL-Felder* bezeichnet. Alle anderen Felder werden *optionale Felder*, *NULL-Felder* oder *Nullable-Felder* genannt.

NOT-NULL-Textfelder

Bei Textfeldern (Felddatentyp `KURZER TEXT` oder `LANGER TEXT`) gibt es übrigens einen Unterschied zwischen `NULL` und *leerer Zeichenfolge* (`""`). Leider werden in Access beide Inhalte in der Tabellenansicht identisch als leerer Kasten angezeigt.

In den meisten Fällen wird eine leere Zeichenfolge nicht aussagekräftig sein. Daher empfehle ich Ihnen, leere Zeichenfolgen für Textfelder zu verbieten. Öffnen Sie dazu die Tabelle in der Entwurfsansicht, und wählen Sie im Registerblatt ALLGEMEIN unter LEERE ZEICHENFOLGE den Wert NEIN aus. Anschließend können Sie über EINGABE ERFORDERLICH festlegen, ob das Textfeld ein NOT-NULL-Feld oder ein optionales Feld ist. Ein Beispiel finden Sie in den Materialien zum Buch unter *02_Access_als_Datenbank_Tabellen\2.3.12_Fluege_leere_Zeichenfolge.accdb*.

FlugID	AbflugDatum	AbflugZeit	AnkunftsZeit	Fluggesellschaft	Flugnummer
1	03.07.2019	15:00:00	16:20:00	easyJet	EJU5846
2	03.07.2019	15:10:00	16:40:00	Austrian Airlines Group	OS131
3	03.07.2019	15:40:00	17:20:00		AF2334
4	03.07.2019	15:50:00	17:35:00	Air France	AF2035
5	03.07.2019	15:55:00	17:15:00	Air France	AF1919

Abbildung 2.26 In diesem Beispiel ist »Fluggesellschaft« ein NOT-NULL-Feld. Im dritten Datensatz enthält das Feld eine leere Zeichenfolge, was keinen Deut besser ist als NULL. Mit der Feldeinschränkung »Leere Zeichenfolge« können Sie das verhindern.

Ich empfehle Ihnen, NOT-NULL-Felder intensiv zu nutzen. Viele Felder werden natürlich optional bleiben müssen. Aber schauen Sie bitte genau hin, ob Ihr Geschäftsmodell das benötigt. Warum liebe ich NOT-NULL-Felder so sehr? Weil optionale Felder Mehraufwand erzeugen. In Abfragen und spätestens bei der Programmierung müssen Sie für jedes optionale Feld den separaten Fall unterscheiden, dass das Feld leer ist. Dieser Mehraufwand tritt bei jeder Art von Fallunterscheidung auf. NOT-NULL-Felder sind hingegen einfacher: Sie können ganz sicher sein, dass überhaupt ein Wert eingetragen ist. Den NULL-Wert (nicht definiert) brauchen Sie für NOT-NULL-Felder nicht zu berücksichtigen.

Ternäre Logik

Mit NULL-Werten kommen Sie bei Fallunterscheidungen mit der *ternären Logik* in Berührung. In der *klassischen Logik*, der *zweiwertigen Logik*, gibt es nur die Werte »wahr« und »falsch«. In der ternären Logik gibt es die Werte »wahr«, »falsch« und »nicht definiert« (NULL). Der Umgang mit der ternären Logik ist aufwendiger. Wenn alle Felder einer Fallunterscheidung NOT-NULL-Felder sind, vereinfacht sich die Logik zur zweiwertigen Logik.

Feld-Gültigkeitsregel

Mit Feld-Gültigkeitsregeln bietet Access eine elegante Möglichkeit, den Inhalt eines Feldes noch weiter einzuschränken. Als Gültigkeitsregel wird eine Formel eingetragen, die Sie ent-

weder direkt eintippen oder mit Hilfe des Ausdrucks-Generators erstellen können. Formeln haben Sie bereits in Abschnitt 2.3.11, »Berechnete Felder«, kurz kennengelernt. An dieser Stelle beschäftigen wir uns mit Formeln für den Vergleich.

Hier ein paar Beispiele für Textfelder:

- ▶ Das Feld soll immer den Wert »München« haben:
= 'München'
- ▶ Alle Werte mit Ausnahme von »München« sind erlaubt:
<> 'München'
- ▶ Als Wert ist entweder »München« oder »Berlin« erlaubt:
'München' Oder 'Berlin'
- ▶ Alle gültigen Werte sind in einer Liste aufgeführt:
In ('Berlin'; 'Hamburg'; 'München')
- ▶ Alle Werte sind erlaubt mit Ausnahme derer aus der Liste:
Nicht In ('Berlin'; 'Hamburg'; 'München')
- ▶ Alle Werte, die mit »M« beginnen, sind erlaubt:
Wie 'M*'
- ▶ Nur Werte, die nicht auf »burg« enden, sind erlaubt.
Nicht Wie '*burg'

Übrigens spielt es keine Rolle, ob Sie Zeichenfolgen mit einfachen Anführungszeichen (') oder doppelten Anführungszeichen (") einrahmen.

Textmustervergleich mit den Operatoren »Wie« und »ALike«

Der Operator **Wie** (englisch *like*) ist ein sehr mächtiges Werkzeug, das einen Text mit einem Muster vergleicht (*Mustervergleich*, englisch *pattern matching*). In Tabelle 2.17 sind alle *Platzhalter* (englisch *wildcard*) mit Beispielen aufgeführt.

ALike ist der zweite Operator zum Textmustervergleich. Die Platzhalter sind jedoch unterschiedlich (siehe Tabelle 2.18). Der Name **ALike** bedeutet »ANSI Like« und lässt vermuten, dass dieser Operator den Vorgaben des ANSI/ISO-SQL-92-Standards entspricht. Wer den Originaltext dieses Standards sorgfältig liest, wird feststellen, dass das nicht der Fall ist! Zutreffender ist eher, dass **ALike** sehr nahe an den **Like**-Operator des Microsoft SQL Servers angelehnt ist.

Ob Sie sich nun für **Like** oder **ALike** entscheiden, ist letztendlich Geschmackssache. Einziger Unterschied sind wie gesagt die Platzhalter. Wenn Sie eine Server-Datenbank verwenden, könnte Ihnen **ALike** besser gefallen.

Platzhalter	Bedeutung	Muster (Beispiel)	passende Zeichenfolgen	nicht passende Zeichenfolgen
*	kein, ein oder mehrere beliebige Zeichen	M*	Mainz, München, M	Hamburg, Berlin, H
?	ein beliebiges Zeichen	H?mburg	Hamburg, Homburg	Mainz, Hmburg, Haamburg
#	eine beliebige Ziffer	LH####	LH2890	LH191, LHxxxx, LH12345
[...]	ein Zeichen aus einer Liste	H[ao]mburg	Hamburg, Homburg	Hemburg, Hmburg, Haamburg
[*] [?] [#] [[[]]	die Sonderzeichen selbst	H[*]mburg	H*mburg	Hamburg, Homburg, Hmburg
[!...]	ein Zeichen, aber nicht aus der Liste	H[!ao]mburg	Hemburg	Hamburg, Homburg, Hmburg, Haamburg
[...!]	ein Zeichen aus einer Liste inklusive Ausrufezeichen	H[ao!]mburg	Hamburg, Homburg, H!mburg	Hemburg, Hmburg
[...-...]	ein Zeichen aus einem Bereich	[a-z]	a, b, c, A, B, ö, ß	aa, 1, 4, %
[!...-...]	ein Zeichen, aber nicht aus einem Bereich	[!0-9]	a, b, c, A, B, %, ö, ß	1, 4, aa

Tabelle 2.17 Platzhalter für den Operator »Wie«

Platzhalter	Bedeutung	Muster (Beispiel)	passende Zeichenfolgen	Nicht passende Zeichenfolgen
%	kein, ein oder mehrere beliebige Zeichen	M%	Mainz, München, M	Hamburg, Berlin, H

Tabelle 2.18 Platzhalter für den Operator »ALike«

Platzhalter	Bedeutung	Muster (Beispiel)	passende Zeichenfolgen	Nicht passende Zeichenfolgen
_	ein beliebiges Zeichen	H_mburg	Hamburg, Homburg	Mainz, Hmburg, Haamburg
[...]	ein Zeichen aus einer Liste	H[ao]mburg	Hamburg, Homburg	Hemburg, Hmburg, Haamburg
[%] [_] [[[]]	die Sonderzeichen selbst	H[%]mburg	H%mburg	Hamburg, Homburg, Hmburg
[!...]	ein Zeichen, aber nicht aus der Liste	H[!ao]mburg	Hemburg	Hamburg, Homburg, Haamburg
[...!]	ein Zeichen aus einer Liste inkl. Ausrufezeichen	H[ao!]mburg	Hamburg, Homburg, H!mburg	Hemburg, Hmburg
[...-...]	ein Zeichen aus einem Bereich	[a-z]	a, b, c, A, B, ö, ß	aa, 1, 4, %
[!...-...]	ein Zeichen, aber nicht aus einem Bereich	[!0-9]	a, b, c, A, B, %, ö, ß	1, 4, aa

Tabelle 2.18 Platzhalter für den Operator »ALike« (Forts.)

Ebenso sind Feld-Gültigkeitsregeln für Zahlenfelder möglich:

- ▶ Nur positive Zahlen:
>0
- ▶ Alle Werte von 1 bis 10 sind erlaubt:
Zwischen 1 Und 10
- ▶ Alle Werte zwischen 1 und 10 (aber 1 und 10 selbst sind nicht dabei):
>1 Und <10
- ▶ Alle Zahlen außer 0:
<>0
- ▶ Eine Liste von Werten zulassen:
In (2;4;8;16)

Aus Abschnitt 2.3.6, »Felddatentyp »Datum/Uhrzeit«, wissen Sie, dass eine Datumsangabe in Access als Zahl abgespeichert wird. Die Vergleiche für Felder mit dem Felddatentyp DATUM/UHRZEIT sehen daher genauso wie die der Zahlenfelder aus:

- Ein Datum ab dem 1. Februar 2019:

>=#01.02.2019#

- Ein Zeitpunkt im Mai 2019:

>=#01.05.2019# Und <#01.06.2019#

- Ein Datum in der Vergangenheit:

<Jetzt()

Auf eine kleine Falle möchte ich Sie hinweisen: Wenn Sie Zeitpunkte z. B. bis zum 31.05.2019 zulassen möchten, vergleichen Sie bitte wie im Beispiel oben mit einem Tag später (dem ersten Juni). Wichtig ist das, wenn Sie in Ihrem Feld Datum und Uhrzeit gleichzeitig erfassen wollen. Würden Sie nämlich <=#31.05.2019# festlegen, dann ist für den 31.05.2019 gerade noch Mitternacht erlaubt. Eine Sekunde später liegt aber schon außerhalb des Gültigkeitsbereichs! Mit dem Vergleich <#01.06.2019# (nicht <=) sind Sie auf der sicheren Seite, und alle Zeitpunkte am letzten Tag bis 23:59:59 Uhr sind gültig.

Angaben von Datum und Uhrzeit in Formeln

Konkrete Zeitpunkte werden in Formeln in einem besonderen Format – eingerahmt durch zwei Hash-Zeichen (#) – eingetragen:

#TT.MM.JJJJ hh:mm:ss#

Möglich sind Angaben zum Datum, zur Uhrzeit oder beides gleichzeitig.

Beispiele:

- der erste Mai 2019: #01.05.2019#
- Zeit für ein Knoppers: #09:30:00#
- Zeit für ein Knoppers am ersten Mai 2019: #01.05.2019 09:30:00#

Sie haben in den Beispielen oben schon gesehen, dass zwei oder mehrere Bedingungen durch die Operatoren Und (englisch And) und Oder (englisch Or) kombiniert werden können. Der Operator Nicht (englisch Not) kehrt die Bedeutung um (*Negation*). Der Vollständigkeit halber erwähne ich noch den Operator ExOder für *exklusives Oder* (englisch *exclusive or*, XOR), das heißt, entweder darf die erste Bedingung wahr sein oder die zweite, aber nicht beide gleichzeitig. Mit diesen Operatoren können Sie umfangreiche Gültigkeitsregeln generieren, beispielsweise:

Nicht -5 Und <0 Oder Nicht 5 Und Zwischen 2 Und 8

Reihenfolge der Operatoren und Klammern

Bei umfangreicheren Formeln wie im letzten Beispiel verliert man manchmal den Überblick, welche Teile der Formel mit welchem Nicht, Und, Oder und ExOder verknüpft sind. Für die Operatoren gilt diese Reihenfolge:

1. Zwischen, In, Wie
2. Nicht
3. Und
4. Oder
5. ExOder
6. =, >, <, >=, <=, <>

Ähnlich wie »Punkt- vor Strichrechnung« gilt also »Nicht vor Und« sowie »Und vor Oder«. Mit runden Klammern können Sie eine abweichende Reihenfolge festlegen. Ich verwende runde Klammern selbst dann, wenn es nicht unbedingt notwendig ist, weil eine Formel nach meinem Empfinden dadurch besser lesbar wird.

Beispiel:

((Nicht -5) Und <0) Oder ((Nicht 5) Und (Zwischen 2 Und 8))

In den Materialien zum Buch finden Sie diese Beispiele in der Datenbank *O2_Access_als_Datenbank_Tabellen\2.3.12_Feldgueltigkeitsregel.accdb*.

Wenn eine Feld-Gültigkeitsregel eingetragen ist, wird Access vor dem Abspeichern den neuen Wert mit Hilfe der Formel prüfen. Wenn die Bedingung wahr ist, wird der Wert abgespeichert. Wenn nicht, erscheint eine Fehlermeldung:

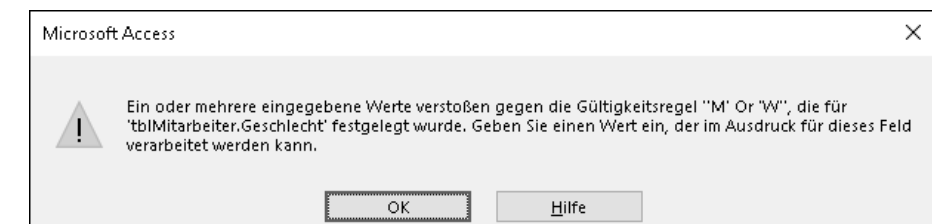


Abbildung 2.27 Ohne eigene Gültigkeitsmeldung erzeugt Access eine Fehlermeldung, in der die Feld-Gültigkeitsregel angezeigt wird. Meiner Erfahrung nach hilft das den wenigsten Anwendern weiter.

Im Registerblatt ALLGEMEIN unter GÜLTIGKEITSMELDUNG können Sie auch eine eigene Fehlermeldung eintragen, die aussagekräftiger ist:

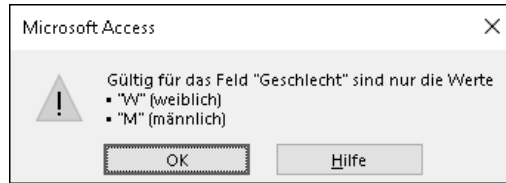


Abbildung 2.28 Hilfreicher und schöner sind eigene Gültigkeitsmeldungen.

Diese Gültigkeitsmeldung finden Sie in den Materialien zum Buch unter *O2_Access_als_Datenbank_Tabellen\2.7.5_Fluege_umfassend.accdb*, dort in der Tabelle *tblMitarbeiter*, Feld »Geschlecht«. Damit Ihre Datenbank richtig schön aussieht, hier die Schritte, mit denen Sie mehrere Zeilen und die Aufzählungspunkte erzeugen:

1. Öffnen Sie die Tabelle *tblMitarbeiter* in der Entwurfsansicht.
2. Wählen Sie das Feld »Geschlecht« aus.
3. Klicken Sie im Registerblatt ALLGEMEIN auf GÜLTIGKEITSMELDUNG.
4. Mit $\text{Ctrl} + \text{F2}$ öffnen Sie das ZOOM-Fenster.
5. Tragen Sie hier den Text ein.
6. Zeilenumbrüche fügen Sie mit $\text{Strg} + \text{↵}$ ein.
7. Aufzählungspunkte erzeugen Sie mit $\text{Alt} + \text{0} \text{ 1} \text{ 4} \text{ 9}$ (Eingabe der Ziffern über den Zahlenblock der Tastatur!)

Standardwert

Wenn Ihre Tabelle viele Felder hat, kann es schon sehr aufwendig werden, einen neuen Datensatz einzutragen. Häufig ist es so, dass einige Feldinhalte immer gleich bleiben. An dieser Stelle können Ihnen *Standardwerte* (englisch *default values*) viel Tipparbeit ersparen. Standardwerte spielen überhaupt nur eine Rolle, wenn ein Datensatz neu erstellt wird.

Als Beispiel möchte ich, dass Access in der Tabelle *tblFlug* in »AbflugFlughafen« standardmäßig immer »Frankfurt am Main« einträgt.

1. Öffnen Sie die Datenbank *O2_Access_als_Datenbank_Tabellen\2.3.12_Fluege.accdb* aus den Materialien zum Buch.
2. Öffnen Sie die Tabelle *tblFlug* in der Entwurfsansicht.
3. Wählen Sie das Feld »AbflugFlughafen« aus.
4. Tragen Sie im Registerblatt ALLGEMEIN unter STANDARDWERT
="Frankfurt am Main"
ein.
5. Klicken Sie in der Symbolleiste für den Schnellzugriff auf SPEICHERN (oder $\text{Strg} + \text{S}$).

Das Ergebnis finden Sie in den Materialien zum Buch unter *O2_Access_als_Datenbank_Tabellen\2.3.12_Fluege_Standardwert.accdb*.

Wenn Sie für ein Feld einen Standardwert festgelegt haben, zeigt Ihnen Access in der Datenblattansicht den Standardwert schon an, bevor Sie den neuen Datensatz begonnen haben.

FlugID	AbflugDatum	AbflugZeit	AnkunftZeit	Fluggesellschaft	Flugnummer	AbflugFlughafen
46	03.07.2019	19:50:00	21:15:00	Air France	AF1819	Frankfurt am Main
47	03.07.2019	20:10:00	21:40:00	Austrian Airlines Group	OS127	Wien
48	03.07.2019	20:15:00	22:30:00	Eurowings	EW357	London Stansted
49	03.07.2019	21:55:00	22:40:00	Ryanair	FR8547	Berlin-Schönefeld
50	04.07.2019	06:05:00	09:40:00	Condor	DE1478	Frankfurt am Main
(Neu)						Frankfurt am Main

Abbildung 2.29 Für den neuen Datensatz (in der letzten Zeile) wird für »AbflugFlughafen« der Standardwert »Frankfurt am Main« angezeigt.

Sie können den Standardwert jederzeit mit Ihrem Wunschwert überschreiben. Insofern sind Standardwerte in Wirklichkeit gar keine Einschränkungen. Sie gehören trotzdem zu den Feldeinschränkungen; zumindest wird das bei der Programmierung per SQL deutlich (mehr dazu in Abschnitt 5.6, »Datenbankobjekte verändern: Data Definition Language (DDL)«).

Ich finde es sehr nützlich, dass auch komplexe Formeln mit Funktionen als Standardwert erlaubt sind. Hier zwei Beispiele für Felder mit dem Felddatentyp DATUM/UHRZEIT:

- ▶ das aktuelle Datum (ohne Uhrzeit):
=Datum()
- ▶ das aktuelle Datum (mit Uhrzeit):
=Jetzt()

An diesen Stellen können Sie Standardwerte in Access festlegen

In Access können Sie Standardwerte an zwei Stellen eintragen:

- ▶ auf Tabellenebene
- ▶ in einem Steuerelement eines Formulars

Um das Thema »Formulare« dreht sich das gesamte Kapitel 6. Das Schöne an Formularen ist, dass Sie für eine Tabelle mehrere verschiedene Formulare erstellen können. Wenn Sie also einen Standardwert für die eine Form der Eingabe benötigen, nicht jedoch in anderen Fällen, dann tragen Sie den Standardwert für das Steuerelement eines Formulars ein.

Noch ein Hinweis: Wenn Sie sowohl auf Tabellenebene als auch im Steuerelement eines Formulars einen Standardwert gesetzt haben, dann hat der Standardwert des Steuerelements im Formular beim Arbeiten im Formular Vorrang.

Zusammenfassung

In diesem Abschnitt haben Sie die Feldeinschränkungen EINGABE ERFORDERLICH, Feld-Gültigkeitsregel und Standardwert kennengelernt. Wenn eine Feldeinschränkung verletzt wird, zeigt Access eine Fehlermeldung an und verhindert, dass der Datensatz gespeichert wird. Mit einer Gültigkeitsmeldung können Sie die Fehlermeldung anpassen. Leider können Sie keine benutzerdefinierte Fehlermeldung für die Verletzung von EINGABE ERFORDERLICH hinterlegen – eigentlich schade. Bei Standardwerten wird keine Regel verletzt, insofern gibt es dafür auch keine Fehlermeldung.

Best Practice für Feldeinschränkungen

- ▶ Setzen Sie EINGABE ERFORDERLICH standardmäßig auf JA.
- ▶ Setzen Sie nur für Felder, die wirklich optional sind, EINGABE ERFORDERLICH auf NEIN.
- ▶ Für Textfelder, für die die Eingabe erforderlich ist, sollten Sie leere Zeichenfolgen nicht zulassen.
- ▶ Formulieren Sie eine Feld-Gültigkeitsregel, wann immer dies möglich ist.
- ▶ Tragen Sie einen Standardwert auf Tabellenebene ein, wenn der Standardwert für alle Formen der Eingabe sinnvoll ist.
- ▶ Tragen Sie einen Standardwert für ein Steuerelement eines Formulars ein, wenn der Standardwert nur für eine bestimmte Form der Eingabe (d. h. ein bestimmtes Formular) sinnvoll ist.

2.3.13 Tabelleneinschränkungen

Im letzten Abschnitt haben wir immer nur den Wert eines einzigen Feldes geprüft. Eine *Tabelleneinschränkung* hingegen kann mehrere Felder gleichzeitig prüfen. Es gibt zwei Arten von Tabelleneinschränkungen. Wir beschäftigen uns zunächst nur mit dem ersten Typ, der *Tabellen-Gültigkeitsregel*.

Mehrere Felder in Kombination prüfen

Verwenden Sie eine Tabellen-Gültigkeitsregel immer dann, wenn mehrere Felder in ihrer Kombination wichtig sind.

Ein Beispiel: Die Tabelle *tblMitarbeiter* speichert für jeden Mitarbeiter die beiden Felder »Geburtsdatum« und »Eintrittsdatum«. Es ist offensichtlich, dass ein Mitarbeiter erst nach seiner Geburt in das Unternehmen eintreten kann. »Eintrittsdatum« muss also später als »Geburtsdatum« liegen.

Diese Art von Regeln können Sie als Tabellen-Gültigkeitsregel definieren. Eine Tabellen-Gültigkeitsregel hängt nie an einem einzelnen Feld, sondern gilt übergreifend für den gesamten Datensatz. Und so erstellen Sie eine Tabellen-Gültigkeitsregel:

1. Öffnen Sie die Datenbank *O2_Access_als_Datenbank_Tabellen\2.3.13_Mitarbeiter.accdb* aus den Materialien zum Buch.
2. Öffnen Sie die Tabelle *tblMitarbeiter* in der Entwurfsansicht (Abbildung 2.30).

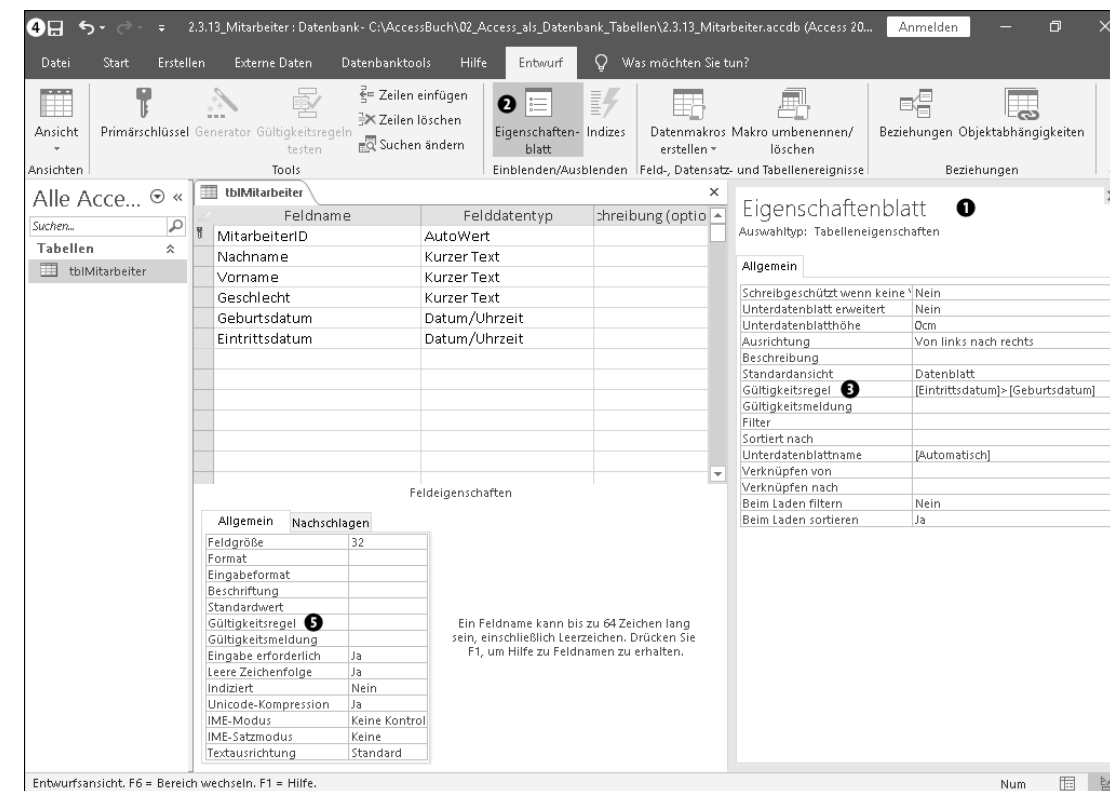


Abbildung 2.30 Eine Tabellen-Gültigkeitsregel können Sie in den Tabelleneigenschaften ❶ unter »Gültigkeitsregel« ❸ festlegen.

3. Falls die Tabelleneigenschaften ❶ noch nicht sichtbar sind: Klicken Sie auf ENTWURF • EIGENSCHAFTENBLATT ❷.
4. Tragen Sie im EIGENSCHAFTENBLATT (Tabelleneigenschaften) ❶ unter GÜLTIGKEITSREGEL ❸
[Eintrittsdatum] > [Geburtsdatum]
ein.
5. Klicken Sie in der Symbolleiste für den Schnellzugriff auf SPEICHERN ❹ (oder **Strg** + **S**).

Wichtig ist, dass Sie diese Gültigkeitsregel an der richtigen Stelle eintragen, nämlich unter GÜLTIGKEITSREGEL in den Tabelleneigenschaften ❶. Nicht jedoch unter ALLGEMEIN • GÜLTIGKEITSREGEL für eines der Datenfelder ❺!

Im Gegensatz zur Feld-Gültigkeitsregel müssen Sie bei der Tabellen-Gültigkeitsregel immer den Feldnamen angeben. Der Feldname wird dazu in eckige Klammern gesetzt ([und]). Mit den Operatoren Und, Oder, ExOder und Nicht können Sie beliebig komplexe Gültigkeitsregeln erzeugen. Hier ein paar Beispiele:

1. Genau eines von drei JA/NEIN-Feldern soll gesetzt sein:

```
[GebundeneAusgabe] ExOder [Broschiert] ExOder [Taschenbuch]
```

2. Nur wenn das JA/NEIN-Feld »IstGestrichen« gesetzt ist, darf und muss in zwei anderen Feldern etwas eingetragen werden:

```
(
  ([IstGestrichen] = Falsch)
  Und ([IstGestrichenWordenDurch] Ist Null)
  Und ([IstGestrichenWordenAm] Ist Null)
)
Oder
(
  ([IstGestrichen] = Wahr)
  Und ([IstGestrichenWordenDurch] Ist Nicht Null)
  Und ([IstGestrichenWordenAm] Ist Nicht Null)
)
```

So weiß jeder, wann und durch welchen Mitarbeiter ein Flug annulliert wurde.

3. Wenn ein Flug verspätet ist, dann muss der tatsächliche Zeitpunkt von Abflug oder Ankunft später als geplant liegen:

```
([IstVerspaetet]=Falsch)
Oder
(
  ([IstVerspaetet]=Wahr)
  Und ([AbflugDatumTatsaechlich] Ist Nicht Null)
  Und ([AbflugZeitTatsaechlich] Ist Nicht Null)
  Und ([AnkunftZeitTatsaechlich] Ist Nicht Null)
  Und
  (
    ([AbflugDatumTatsaechlich]>[AbflugDatumGeplant])
    Oder ([AbflugZeitTatsaechlich]>[AbflugZeitGeplant])
    Oder ([AnkunftZeitTatsaechlich]>[AnkunftZeitGeplant])
  )
)
```

Alle diese Beispiele von Tabellen-Gültigkeitsregeln finden Sie in den Materialien zum Buch in der Datenbank *O2_Access_als_Datenbank_Tabellen\2.3.13_Tabellengueltigkeitsregel.accdb*.

Sie merken schon, dass Tabellen-Gültigkeitsregeln sehr umfangreich werden können. Um überhaupt einen Überblick zu erhalten, welches Und und Oder zu welcher Klammer gehört, habe ich die Tabellen-Gültigkeitsregeln in mehreren Zeilen und mit Einrückungen formatiert. Meistens erledige ich das in einem Texteditor, denn nach dem Abspeichern verwirft Access die Formatierungen und packt alles wieder in eine einzige Zeile.

Mehr als eine Tabellen-Gültigkeitsregel pro Tabelle

Sie können auch mehr als eine Tabellen-Gültigkeitsregel eintragen. Dazu trennen Sie die einzelnen Gültigkeitsregeln mit dem Operator Und:

```
(<Tabellen-Gültigkeitsregel 1>)
Und
(<Tabellen-Gültigkeitsregel 2>)
Und
...
```

Dieses konkrete Beispiel finden Sie in den Materialien zum Buch unter *O2_Access_als_Datenbank_Tabellen\2.3.13_Tabellengueltigkeitsregel.accdb* in der Tabelle *tblFlugMehrereTabellen-Gueltingkeitsregeln*:

```
(
  ([IstVerspaetet]=Falsch)
  Oder
  (
    ([IstVerspaetet]=Wahr)
    Und ([AbflugDatumTatsaechlich] Ist Nicht Null)
    Und ([AbflugZeitTatsaechlich] Ist Nicht Null)
    Und ([AnkunftZeitTatsaechlich] Ist Nicht Null)
    Und
    (
      ([AbflugDatumTatsaechlich]>[AbflugDatumGeplant])
      Oder ([AbflugZeitTatsaechlich]>[AbflugZeitGeplant])
      Oder ([AnkunftZeitTatsaechlich]>[AnkunftZeitGeplant])
    )
  )
)
Und
(
  ([IstGestrichen] = Falsch)
  Und ([IstGestrichenWordenDurch] Ist Null)
  Und ([IstGestrichenWordenAm] Ist Null)
)
```

```
Oder
(
  ([IstGestrichen] = Wahr)
  Und ([IstGestrichenWordenDurch] Ist Nicht Null)
  Und ([IstGestrichenWordenAm] Ist Nicht Null)
)
)
```

Eine Tabellen-Gültigkeitsregel prüfen

Bei umfangreichen Tabellengültigkeitsregeln passieren schnell kleinere Fehler, die nur schwer zu finden sind. Ich empfehle Ihnen, die Tabellen-Gültigkeitsregel entweder ganz oder in Teilen mit Beispieldaten zu testen. Hier ein Tipp für Profis, die schon Erfahrung im Umgang mit Abfragen haben:

1. Erstellen Sie die Tabelle ohne Tabellen-Gültigkeitsregel.
2. Tragen Sie Testdaten in die Tabelle ein (sowohl konsistente als auch inkonsistente Datensätze!).
3. Erstellen Sie eine neue Abfrage in der SQL-Ansicht.
4. So finden Sie alle konsistenten Datensätze:

```
SELECT *
FROM <Tabellenname>
WHERE (<Tabellen-Gültigkeitsregel>);
```

5. Und so finden Sie alle inkonsistenten Datensätze:

```
SELECT *
FROM <Tabellenname>
WHERE NOT (<Tabellen-Gültigkeitsregel>);
```

Auf diese Weise können Sie die Tabellen-Gültigkeitsregel schrittweise erstellen und überprüfen. Wenn alles richtig ist, löschen Sie die inkonsistenten Datensätze aus der Tabelle und tragen die fertige Formel als Tabellen-Gültigkeitsregel in der Entwurfsansicht der Tabelle ein.

Eine Tabellen-Gültigkeitsregel bezieht sich immer nur auf einen Datensatz

Mit einer Tabellen-Gültigkeitsregel können Sie immer nur jeden einzelnen Datensatz für sich überprüfen. Oder anders gesagt, Sie können in der Formel für die Tabellen-Gültigkeitsregel nicht auf die Werte in anderen Datensätzen zugreifen.

Ein Beispiel, beim dem eine Tabellen-Gültigkeitsregel *nicht* weiterhilft: Jeder Mitarbeiter in der Tabelle *tblMitarbeiter* bekommt eine Personalnummer, die eindeutig pro Mitarbeiter

sein soll. In sämtlichen Datensätzen von *tblMitarbeiter* darf jede Personalnummer höchstens einmal auftreten.

Mit einer Tabellen-Gültigkeitsregel kommen Sie hier nicht weit! Denn schließlich müssten Sie alle bisher vergebenen Personalnummern, also alle anderen Datensätze in *tblMitarbeiter*, überprüfen. Ich hatte eingangs erwähnt, dass es zwei Arten von Tabelleneinschränkungen gibt. Für diesen Fall könnten Sie den zweiten Typ, nämlich den *eindeutigen Schlüssel*, einsetzen, den ich Ihnen in Abschnitt 2.9.2, »Eindeutiger Schlüssel (Alternativschlüssel)«, vorstellen werde.

Zusammenfassung

Je nach Art der Einschränkung können Sie also nur auf bestimmte Felder einer Tabelle zugreifen:

Einschränkung	Bezieht sich auf ein einziges Feld	Kann auf alle Felder in einem Datensatz zugreifen	Kann alle Werte der gesamten Tabelle überprüfen
Felddatentyp	•		
EINGABE ERFORDERLICH	•		
Feld-Gültigkeitsregel	•		
Standardwert	•		
Tabellen-Gültigkeitsregel		•	
eindeutiger Schlüssel			•

Tabelle 2.19 Eine Übersicht der verschiedenen Einschränkungen mit Zugriffsbereich

Sie können jede Feld-Gültigkeitsregel als Tabellen-Gültigkeitsregel darstellen. Die Tabellen-Gültigkeitsregel ist somit die allgemeinere Einschränkung verglichen mit einer Feld-Gültigkeitsregel. Beispielsweise wäre für das Feld »Geschlecht« in der Tabelle *tblMitarbeiter* die Feld-Gültigkeitsregel

```
'M' Oder 'W'
```

sinngemäß identisch mit der Tabellen-Gültigkeitsregel

```
[Geschlecht]='M' Oder [Geschlecht]='W'
```

Gleichwohl rate ich Ihnen davon ab, denn die Feld-Gültigkeitsregel ist schneller eingetippt. Vor allem aber hängt sie an dem Feld, auf das sie sich bezieht, was für meinen Geschmack die Lesbarkeit und Verständlichkeit ungemein erhöht.

Best Practice für Tabellen-Gültigkeitsregeln

- ▶ Tragen Sie eine Gültigkeitsregel, die sich nur auf ein Feld bezieht, als Feld-Gültigkeitsregel ein.
- ▶ Definieren Sie eine Gültigkeitsregel, die sich auf mehrere Felder bezieht, als Tabellen-Gültigkeitsregel.
- ▶ Tabellen-Gültigkeitsregeln werden recht schnell umfangreich und komplex. Hier gilt es abzuwägen, was unbedingt notwendig ist und welche Regeln nicht so wichtig sind.
- ▶ Bauen Sie umfangreiche Tabellen-Gültigkeitsregeln schrittweise auf.
- ▶ Prüfen Sie Tabellen-Gültigkeitsregeln mit Beispieldaten.
- ▶ Wenn Sie sehr viele Regeln prüfen müssen, kann Programmcode in einem Formular übersichtlicher sein.

2.3.14 Der Nachschlage-Assistent

Neben den besprochenen Felddatentypen führt Access im Dropdown-Menü FELDDATENTYP zusätzlich die Option NACHSCHLAGE-ASSISTENT ... auf. Dies ist *kein* eigener Felddatentyp, sondern erzeugt ein Textfeld mit Hilfe eines Assistenten. Der Assistent befüllt letztendlich die Angaben im Registerblatt NACHSCHLAGEN, auf das ich in Abschnitt 2.6.4, »Nachschlagen«, eingehen werde. Da der Assistent weitgehend selbsterklärend ist, werde ich ihn – wie üblich in diesem Buch – nicht näher erläutern.

2.4 Ändern von Tabellen

In vorherigen Abschnitt habe ich Ihnen gezeigt, wie Sie eine Tabelle in der Entwurfsansicht öffnen und dort die Felder festlegen. Damit kennen Sie bereits das Werkzeug, mit dem Sie eine Tabelle ändern, recht gut: die *Entwurfsansicht*. In diesem Abschnitt werde ich Ihnen zeigen, wie Sie die Felder einer Tabelle (die *Tabellenstruktur*) ändern können, wenn bereits Datensätze in der Tabelle vorhanden sind.

2.4.1 Hinzufügen von Feldern

Ein neues Feld zu einer bestehenden Tabelle hinzuzufügen ist eigentlich eine langweilige Sache:

1. Öffnen Sie die Tabelle in der Entwurfsansicht.
2. Gehen Sie in die leere Zeile nach dem letzten Feld.

3. Legen Sie den Namen des Feldes unter FELDNAME fest.
4. Anschließend wählen Sie unter FELDDATENTYP den passenden Felddatentyp aus.
5. Optional stellen Sie weitere Eigenschaften zum Felddatentyp, zu Feldeinschränkungen und zu Formatierungen im Registerblatt ALLGEMEIN ein.
6. Klicken Sie in der Symbolleiste für den Schnellzugriff auf SPEICHERN (oder `Strg` + `S`).

Wenn in Ihrer Tabelle Datensätze vorhanden sind, dann hat es der letzte Schritt aber in sich! In der Datenbankdatei (.accdb- oder .mdb-Datei) werden die Datensätze hintereinander weggespeichert, so wie sie einmal eingegeben wurden. Ein neues Feld bedeutet, dass Access Platz schaffen muss. Und das betrifft alle schon vorhandenen Datensätze. Gerade bei umfangreichen Tabellen kann das Umkopieren der Daten sehr lange dauern.

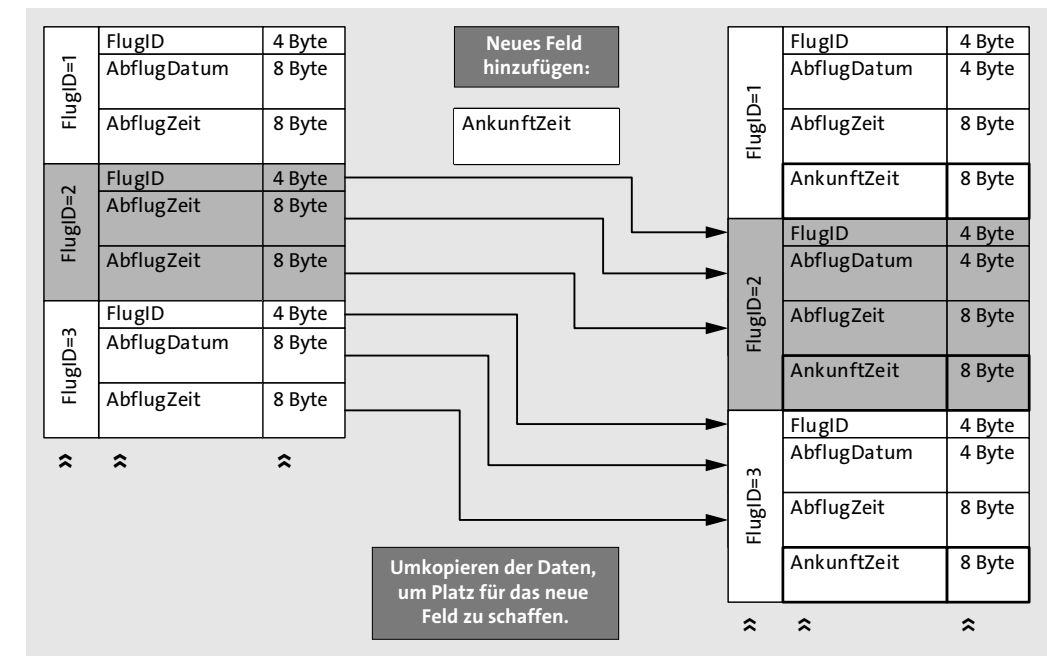


Abbildung 2.31 Wenn Sie ein neues Feld in die Tabelle »tblFlug« einfügen, müssen die bestehenden Datensätze umkopiert werden, um Platz für das neue Feld zu schaffen.

Für bestehende Datensätze bekommt das neue Feld den Wert NULL. Meiner Erfahrung nach erledigt Access das Umkopieren der Daten recht zuverlässig. Ich empfehle Ihnen außerdem, in regelmäßigen Abständen die Datenbank zu komprimieren (DATENBANKTOOLS • DATENBANK KOMPRIMIEREN UND REPARIEREN). Diese Funktion bewirkt, dass alle Tabellen (und andere Datenbankobjekte) umkopiert werden. Dabei werden die Datensätze optimal angeordnet.

2.4.2 Verschieben von Feldern

In der Entwurfsansicht einer Tabelle können Sie die Reihenfolge der Felder verändern.

Feldname	Felddatentyp
FlugID	AutoWert
AbflugDatum	Datum/Uhrzeit
AbflugZeit	Datum/Uhrzeit
AnkunftsZeit	Datum/Uhrzeit
Fluggesellschaft	Kurzer Text
Flugnummer	Kurzer Text
AbflugFlughafen	Kurzer Text
AnkunftFlughafen	Kurzer Text
Passagiere	Kurzer Text

Abbildung 2.32 Felder können Sie markieren, indem Sie auf das leere Feld **1** vor dem Feldnamen klicken.

1. Klicken Sie auf das leere Feld vor dem Feldnamen **1**. Das Feld wird dadurch ausgewählt.
2. Ziehen Sie das Feld per Drag & Drop an die gewünschte Position.
3. Klicken Sie in der Symbolleiste für den Schnellzugriff auf **SPEICHERN** (oder **Strg** + **S**).

Auch hier gilt wieder: Sobald Datensätze in der Tabelle vorhanden sind, beginnt Access beim Speichern mit dem Umkopieren der Datensätze. Bei umfangreichen Tabellen kann also eine kleine Drag-and-Drop-Operation eine ganze Menge Kopierarbeit erzeugen.

2.4.3 Löschen von Feldern

So entfernen Sie ein Feld, das Sie nicht mehr benötigen:

1. Öffnen Sie die Tabelle in der Entwurfsansicht.
2. Wählen Sie das Feld aus, das entfernt werden soll (klicken Sie dazu das leere Feld vor dem Feldnamen an).
3. Drücken Sie **Entf**.
4. Klicken Sie in der Symbolleiste für den Schnellzugriff auf **SPEICHERN** (oder **Strg** + **S**).

Wenn Datensätze vorhanden sind, gehen die gelöschten Felder unwiederbringlich verloren. Darauf weist Sie die Meldung aus Abbildung 2.33 hin:

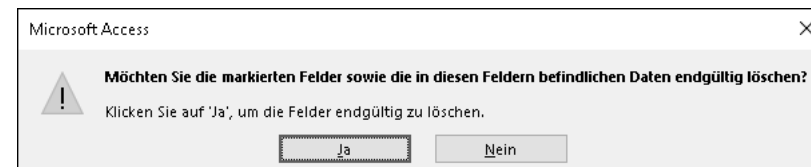


Abbildung 2.33 Beim Drücken auf »Entf« warnt Access, dass die Daten in den gelöschten Feldern verlorengehen werden. Wirklich gelöscht wird das Feld jedoch erst, wenn Sie die Änderungen abspeichern.

Beginnt Access beim Löschen von Feldern auch mit dem Umkopieren bestehender Datensätze, ähnlich wie beim Hinzufügen oder Verschieben von Feldern? Interessanterweise kopiert Access in diesem Fall *nicht* um! Ich habe einmal eine Miniversion unserer Tabelle *tblFlug* mit diesen Feldern eingerichtet:

- ▶ »FlugID«
- ▶ »Flugnummer«
- ▶ »AbflugFlughafen«
- ▶ »AnkunftFlughafen«

Anschließend habe ich drei Datensätze eingetragen und dann das letzte Feld »AnkunftFlughafen« gelöscht. In der Datenblattansicht sieht *tblFlug* jetzt so aus:

FlugID	Flugnummer	AbflugFlughafen
1	LH401	New York
2	AF1919	Frankfurt
3	LH191	Berlin Tegel
*	(Neu)	

Abbildung 2.34 Die Tabelle »tblFlug« mit drei Datensätzen, nachdem ich das letzte Feld »AnkunftFlughafen« gelöscht habe

Lassen Sie uns einen Blick in die Datenbankdatei werfen. Dies ist ein Ausschnitt aus der Datenbankdatei in hexadezimaler Darstellung und als Text:

```
00053f70 00 00 00 00 00 00 00 00 00 04 00 03 00 00 ff .....ÿ
00053f80 fe 4c 48 31 39 31 ff fe 42 65 72 6c 69 6e 20 54 bLH191ÿpBerlin T
00053f90 65 67 65 6c ff fe 4d fc 6e 63 68 65 6e 24 00 1b egelypMünchen$.
00053fa0 00 0d 00 06 00 03 00 0f 04 00 02 00 00 ff fe .....ÿp
00053fb0 41 46 31 39 31 39 ff fe 46 72 61 6e 6b 66 75 72 AF1919ÿpFrankfur
00053fc0 74 ff fe 50 61 72 69 73 20 00 19 00 0e 00 06 00 tÿpParis .....
00053fd0 03 00 0f 04 00 01 00 00 00 ff fe 4c 48 34 30 31 .....ÿpLH401
00053fe0 ff fe 4e 65 77 20 59 6f 72 6b ff fe 46 72 61 6e ÿpNew YorkÿpFran
00053ff0 6b 66 75 72 74 22 00 17 00 0d 00 06 00 03 00 0f kfurtÿp.....
00054000 04 01 05 0e 51 00 00 00 00 00 00 00 00 00 00 .....Q.....
```

Abbildung 2.35 So ist die Tabelle »tblFlug« in der Datenbankdatei gespeichert: Die Inhalte des vierten, mittlerweile gelöschten Feldes »AnkunftFlughafen« sind immer noch vorhanden!

Die etwas kryptischen Zeichen verwendet Access für interne Zwecke, beispielsweise um die Felder zu trennen. Aber Sie sehen sofort: Alle drei Anknüpfungsflughäfen stehen immer noch in der Datei! Leider gibt es in Access die Funktion »Feld löschen rückgängig machen« nicht; diese Felddaten sind also unbrauchbar.

Beim Löschen eines Feldes geht Access sparsam vor: Es wird kein neuer Platz gebraucht, also wird auch nicht umkopiert. Verlassen würde ich mich darauf aber ehrlich gesagt nicht. Auch an dieser Stelle rate ich Ihnen: Komprimieren Sie Ihre Datenbank immer mal wieder (DATENBANKTOOLS • DATENBANK KOMPRIMIEREN UND REPARIEREN). Diese Funktion entfernt dann auch wirklich die Altlasten aus gelöschten Feldern.

2.4.4 Ändern des Felddatentyps

Sie können den Felddatentyp eines bestehenden Feldes nachträglich ändern:

1. Öffnen Sie die Tabelle in der Entwurfsansicht.
2. Gehen Sie in das Feld, das geändert werden soll.
3. Wählen Sie unter FELDDATENTYP den neuen Felddatentyp aus.
4. Klicken Sie in der Symbolleiste für den Schnellzugriff auf SPEICHERN (oder **Strg** + **S**).

Wie Sie sicherlich schon ahnen, wird Access auch hier ein Umkopieren bestehender Datensätze starten, nachdem Sie die Änderungen gespeichert haben. Zusätzlich wird Access versuchen, die bestehenden Daten in den neuen Felddatentyp umzuwandeln. Das wird aber nicht immer gelingen! Eine Konvertierung von DATUM/UHRZEIT in Text wird fehlerfrei ablaufen, wenn das Textfeld entsprechend lang ist. Aber umgekehrt sieht es ganz anders aus. Wenn in einem Textfeld kein gültiges Datum steht, schlägt die Konvertierung fehl, und NULL wird als neuer Wert eingetragen. Ähnliches kann Ihnen beim Konvertieren von Text in ZAHL oder WÄHRUNG passieren.

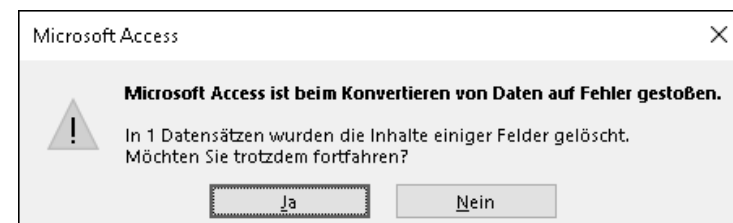


Abbildung 2.36 Beim Ändern des Felddatentyps versucht Access, die Inhalte zu konvertieren. Wenn dies nicht gelingt, wird NULL eingetragen.

2.4.5 Ändern von Feld- und Tabelleneinschränkungen

Auch Feld- und Tabelleneinschränkungen können Sie nachträglich ändern:

1. Öffnen Sie die Tabelle in der Entwurfsansicht.
2. Tragen Sie die neue Tabelleneinschränkung im EIGENSCHAFTENBLATT (Tabelleneigenschaften) unter GÜLTIGKEITSREGEL ein.
3. Gehen Sie in das Feld, das geändert werden soll.
4. Tragen Sie die neue Feldeinschränkung im Registerblatt ALLGEMEIN unter GÜLTIGKEITSREGEL ein.
5. Optional können Sie im Registerblatt ALLGEMEIN weitere Anpassungen vornehmen, beispielsweise EINGABE ERFORDERLICH ändern oder den STANDARDWERT anpassen.
6. Klicken Sie in der Symbolleiste für den Schnellzugriff auf SPEICHERN (oder **Strg** + **S**).

Wenn die Tabelle bereits Daten enthält, können Sie entscheiden, ob Access die Konsistenz der Daten überprüfen soll:

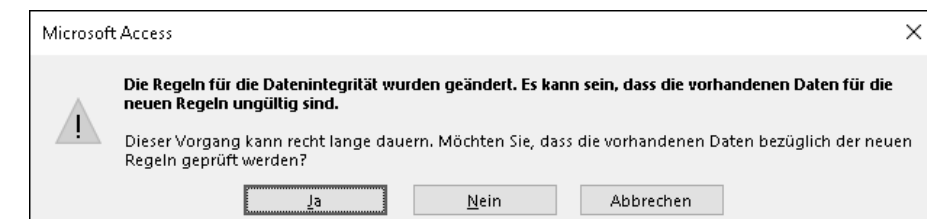


Abbildung 2.37 Nach Ändern einer Feld- oder Tabelleneinschränkung sollten Sie überprüfen lassen, ob die bestehenden Daten immer noch konsistent sind.

Ich empfehle Ihnen, diese Überprüfung durchzuführen. Nur so erfahren Sie, ob in der Tabelle inkonsistente Daten vorhanden sind.

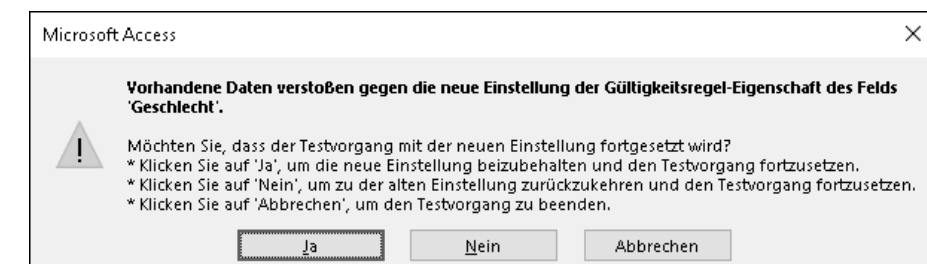


Abbildung 2.38 Access hat einige Datensätze gefunden, die gegen die neue Gültigkeitsregel verstoßen.

Inkonsistente Datensätze werden bei dieser Prüfung von Access weder gelöscht noch verändert. Wenn Sie die Fehlermeldung aus Abbildung 2.38 sehen, müssen Sie also selbst Hand anlegen und die fehlerhaften Datensätze anpassen!

Best Practice beim Ändern von Feld- und Tabelleneinschränkungen

Ich empfehle Ihnen die folgende Vorgehensweise, um inkonsistente Datensätze zu vermeiden:

1. Erstellen Sie eine neue Gültigkeitsregel (z. B. in einem Texteditor).
2. Tragen Sie die neue Gültigkeitsregel in die Entwurfsansicht der Tabelle ein.
3. Prüfen Sie die Datenintegrität.
4. Falls eine Inkonsistenz gemeldet wurde, klicken Sie auf NEIN und verwerfen die neue Gültigkeitsregel zunächst einmal.
5. Suchen Sie jetzt die inkonsistenten Datensätze (entweder manuell oder über eine Abfrage).
6. Passen Sie die inkonsistenten Datensätze so an, dass sie der neuen Gültigkeitsregel entsprechen.
7. Gehen Sie zurück zu Schritt 2.

2.5 Daten in Tabellen

In diesem Abschnitt zeige ich Ihnen, wie Sie Daten in Tabellen eingeben und ändern. In den letzten Abschnitten haben Sie das Werkzeug der Entwurfsansicht kennengelernt, mit dem Sie die Tabellenstruktur festlegen können (Felder, Feldeinschränkungen, Tabelleneinschränkungen). In diesem Abschnitt stelle ich Ihnen das zweite wichtige Tabellen-Werkzeug vor: die *Datenblattansicht*. Für diese Arbeiten können Sie die Datenblattansicht verwenden:

- ▶ Datensätze lesen
- ▶ Datensätze ändern
- ▶ Datensätze löschen
- ▶ Datensätze filtern
- ▶ Datensätze sortieren

Auch für Abfragen gibt es die Datenblattansicht, die dort exakt genauso funktioniert, wie ich Ihnen in den nächsten Abschnitten zeigen werde. Mit Abfragen werden wir uns ausführlich in Kapitel 3, »Daten filtern, sortieren und zusammenfassen: Abfragen«, beschäftigen.

2.5.1 Daten eingeben, ändern und löschen

In der Datenblattansicht sehen Sie die Felder als Spalten. Zur Erinnerung: In der Entwurfsansicht wurden die Felder in Zeilen dargestellt. Beim Umschalten in die Datenblattansicht hat Access die Anordnung der Felder um 90° gedreht. Jede Zeile der Datenblattansicht wird als *Datensatz* bezeichnet. Das leere, graue Feld in der ersten Spalte ist der *Datensatzmarkierer*, der uns – wie Sie gleich sehen werden – nützliche Informationen anzeigt.

Einen neuen Datensatz anlegen

Die letzte Zeile der Tabelle steht für den neuen Datensatz. Sie erkennen dies am Sternchen im Datensatzmarkierer. Bei längeren Tabellen gelangen Sie zum neuen Datensatz, indem Sie auf **START • GEHE ZU • NEU** klicken (oder **Strg** + **+**).

FlugID	AbflugDatum	AbflugZeit
48	03.07.2019	20:15:00
49	03.07.2019	21:55:00
50	04.07.2019	06:05:00
*	(Neu)	

Abbildung 2.39 In der letzten Zeile markiert das Sternchen den neuen Datensatz.

Der neue Datensatz existiert noch nicht, sondern wird Ihnen nur als Zeile zur Eingabe angezeigt. In Feldern, für die ein Standardwert gesetzt ist, wird der neue Wert bereits angezeigt. Im **AUTOWERT**-Feld wird immer (NEU) angezeigt. In dem Moment, wo Sie mit der Eingabe von Daten beginnen, wird der Datensatz erzeugt (aber noch nicht abgespeichert).

Der neue Datensatz wird nur angezeigt, wenn Sie über ausreichende Berechtigungen verfügen

Beim Anzeigen einer Tabelle in der Datenblattansicht überprüft Access, ob Sie überhaupt einen neuen Datensatz erstellen dürfen. In den folgenden Fällen ist das nicht möglich:

- ▶ Die Datenbankdatei ist schreibgeschützt.
- ▶ Ihnen fehlen Dateiberechtigungen zum Ändern der Datenbankdatei.
- ▶ Bei einer Server-Datenbank fehlen Zugriffsberechtigungen zum Erstellen eines neuen Datensatzes.

In diesen Fällen zeigt Access den neuen Datensatz *nicht* an.

Daten in Felder eingeben

Zum Eingeben von Daten wählen Sie das gewünschte Feld aus und beginnen mit dem Tippen. Mit **Tab** oder **↵** gelangen Sie in das nächste Feld, per **↶** + **Tab** in das vorherige.

Tastenkombination	Funktion
Strg + +	zum neuen Datensatz springen
Tab , ↵ oder →	in das nächste Feld springen
↶ + Tab oder ↶	in das vorherige Feld springen

Tabelle 2.20 Tastenkombinationen in der Datenblattansicht

Tastenkombination	Funktion
[Pos1]	zum ersten Feld springen
[Ende]	zum letzten Feld springen
[Strg] + [↵]	Zeilenumbruch einfügen
[Strg] + [#]	Wert aus dem vorherigen Datensatz übernehmen
[Strg] + [;]	das aktuelle Datum einfügen
[Strg] + [:]	die aktuelle Uhrzeit einfügen
[⇧] + [F2]	ZOOM-Fenster aufrufen
[↑]	zum aktuellen Feld im vorherigen Datensatz wechseln
[Strg] + [↑]	zum aktuellen Feld im ersten Datensatz wechseln
[Strg] + [Pos1]	zum ersten Feld im ersten Datensatz wechseln
[↓]	zum aktuellen Feld im nächsten Datensatz wechseln
[Strg] + [↓]	zum aktuellen Feld im letzten Datensatz wechseln
[Strg] + [Ende]	zum letzten Feld im letzten Datensatz wechseln
[⇧] + [↵]	Datensatz speichern
[Entf] oder [Strg] + [-]	Datensatz löschen
[Esc]	Änderungen abbrechen
[Strg] + [Z]	Änderungen rückgängig machen

Tabelle 2.20 Tastenkombinationen in der Datenblattansicht (Forts.)

Das Verhalten der Tasten »Tab« und »Eingabe« anpassen

Rufen Sie dazu über DATEI • OPTIONEN das Fenster ACCESS-OPTIONEN auf. Unter CLIENTEINSTELLUNGEN • BEARBEITEN können Sie die Dateneingabe konfigurieren.

Sehr hilfreich empfinde ich das ZOOM-Fenster, das Sie mit [⇧] + [F2] aufrufen können. Hier lässt sich längerer Text in einem Textfeld besser anzeigen und eintragen. Zeilenumbrüche fügen Sie mit [Strg] + [↵] ein.

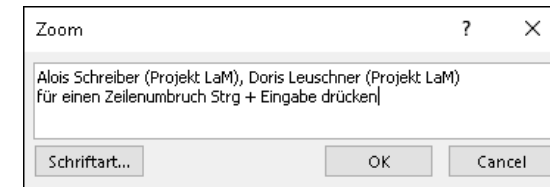


Abbildung 2.40 Von jedem Feld aus können Sie das »Zoom«-Fenster per [⇧] + [F2] aufrufen.

Einen Datensatz speichern

Sobald Sie eine Eingabe tätigen, wird der Datensatz in den *Editiermodus* (auch genannt: *Bearbeitungsmodus*) geschaltet. Sie erkennen dies am Stift im Datensatzmarkierer.

FlugID	AbflugDatum	AbflugZeit	Ankunftszeit
1	03.07.2019	12:00:00	16:20:00
2	03.07.2019	15:10:00	16:40:00
3	03.07.2019	15:40:00	17:20:00
4	03.07.2019	15:50:00	17:35:00
5	03.07.2019	15:55:00	17:15:00

Abbildung 2.41 Der Stift im Datensatzmarkierer kennzeichnet den Editiermodus: Der Datensatz wird gerade verändert, ist aber noch nicht gespeichert.

Zum Abspeichern des Datensatzes haben Sie mehrere Möglichkeiten:

- ▶ Klicken Sie auf den Stift im Datensatzmarkierer.
- ▶ Klicken Sie auf START • DATENSÄTZE • SPEICHERN (oder [⇧] + [↵]).
- ▶ Gehen Sie zu einem anderen Datensatz, beispielsweise per Mausklick in eine andere Zeile.

Vor dem Abspeichern eines Datensatzes überprüft Access, ob alle Feld- und Tabelleneinschränkungen eingehalten werden. Wenn eine Einschränkung verletzt wird, zeigt Access eine Fehlermeldung an. In diesem Fall wird der Datensatz *nicht abgespeichert*, und er bleibt weiterhin im Editiermodus.

Aus den anderen Office-Programmen kennen Sie das Diskettensymbol, mit dem Sie Änderungen abspeichern können. In Access hat diese Schaltfläche eine andere Bedeutung! Änderungen an einem Datensatz können Sie wie oben beschrieben speichern. Verwenden Sie hingegen die Schaltfläche mit der Diskette (oder [Strg] + [S]), wenn Sie Änderungen an der *Tabellenstruktur* oder am *Layout der Tabelle* (z. B. Breite der Spalten) abspeichern möchten.

Änderungen an einem Datensatz abbrechen

Schließlich gibt es noch die Funktion, den Editiermodus abzubrechen. Wenn Sie gerade etwas in ein Feld eintragen, können Sie die Änderungen mit [Esc] rückgängig machen. Wenn Sie noch einmal [Esc] drücken, brechen Sie das Editieren des Datensatzes ab.

Ebenso gibt es die Schaltfläche RÜCKGÄNGIG (englisch *undo*), mit der Sie die letzten Änderungen an einem Datensatz rückgängig machen. Im Gegensatz zu den anderen Office-Programmen können aber nur sehr wenige Schritte rückgängig gemacht werden.

In Datenbanken kein Undo!

Schon mit einer einzelnen Operation können Sie in einer Datenbank große Datenmengen verändern. Denken Sie einmal an Operationen wie »alle Datensätze löschen«. In Abschnitt 3.5, »Aktionsabfragen«, werde ich ein weiteres Verfahren besprechen, mit dem Sie Daten im großen Stil verändern können.


Eine Datenbank kann gleichzeitig durch mehrere Benutzer geöffnet sein (*Multi-User-Funktionalität*). Jeder der Benutzer kann gleichzeitig Daten eingeben, ändern und löschen – sogar an ein und derselben Tabelle!

Diese beiden Fakten bewirken, dass Datenbanken schon an sich recht komplex sind. Eine umfangreiche Rückgängig-Funktion mit mehreren Stufen ist hier nur schwer zu realisieren. Vor allem aber könnte sie zu Verwirrungen führen. Denn immer dann, wenn Sie eine Änderung an einem Datensatz abgespeichert haben, sehen alle anderen Benutzer die neuen Daten *sofort* und arbeiten damit unter Umständen schon weiter.

An einem Beispiel wird das deutlicher: Wenn Sie einen Mitarbeiter auf einen Flug buchen, dann ist der Platz im Flieger belegt, und Kosten für das Ticket sind entstanden. Diese Buchung zu stornieren, will jetzt gut überlegt sein: Hat inzwischen einer Ihrer Kollegen mit den Buchungsdaten bereits weitergearbeitet? Verwenden Sie also die Schaltfläche RÜCKGÄNGIG mit Bedacht!

Gerade in Datenbanken, die von mehreren Benutzern gleichzeitig verwendet werden, empfehle ich Ihnen, sich diesen Merksatz einzuprägen: »In Datenbanken kein Undo!« Sie können eine Flugbuchung stornieren, aber das wäre ein eigener Vorgang. Und zwar ein Vorgang, der mit Ihren Kollegen abgestimmt sein muss.

Einen Datensatz löschen

Um einen Datensatz zu löschen, müssen Sie ihn zuerst markieren. Klicken Sie dazu auf den Datensatzmarkierer. Sie können auch mehrere Datensätze markieren, indem Sie die - oder **[Strg]**-Taste gedrückt halten. Zum Löschen drücken Sie anschließend auf **[Entf]** oder **[Strg] + [Del]**.

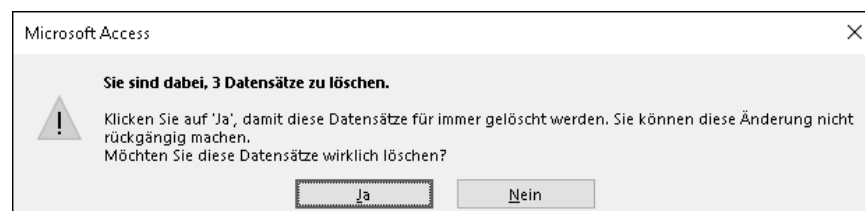


Abbildung 2.42 Vor dem Löschen von Datensätzen zeigt Access diese Warnmeldung an.

Wenn Sie die Warnmeldung mit JA bestätigen, sind die Daten wirklich weg. Beim Löschen von Datensätzen ist die Funktion RÜCKGÄNGIG nicht verfügbar (in Datenbanken kein Undo!).

2.5.2 Navigationsschaltflächen

Am unteren Rand der Datenblattansicht finden Sie die *Navigationsschaltflächen*.



Abbildung 2.43 Die Navigationsschaltflächen am unteren Rand der Datenblattansicht sind besonders hilfreich beim Navigieren in umfangreichen Tabellen.

Mit diesem Werkzeug können Sie zum ersten **1**, vorherigen **2**, nächsten **4**, letzten **5** oder einem neuen Datensatz **6** wechseln. Im Textfeld **3** können Sie auch direkt die Nummer des Datensatzes, zu dem Sie springen möchten, eintippen. Gemeint ist hier wirklich die Zeilennummer, nicht die ID des Primärschlüssels. Diese Schaltflächen zur Navigation in den Datensätzen sind vor allem bei umfangreichen Tabellen sehr hilfreich.

2.5.3 Die Spaltenbreite eines Feldes anpassen

Wie üblich in Office-Programmen können Sie die Spaltenbreite verändern, indem Sie zwischen zwei Spalten klicken und die Spaltenbreite durch Ziehen mit der Maus anpassen.

1. Bewegen Sie dazu den Mauszeiger auf dem Spaltentitel genau auf die vertikale Trennlinie zwischen zwei Spalten. Der Mauszeiger ändert sich wie in Abbildung 2.44 dargestellt:

FlugID	AbflugDatum	AbflugZeit	AnkunftZeit
1	03.07.2019	15:00:00	16:20:00
2	03.07.2019	15:10:00	16:40:00
3	03.07.2019	15:40:00	17:20:00
4	03.07.2019	15:50:00	17:35:00

Abbildung 2.44 Verändern Sie die Spaltenbreite, indem Sie genau zwischen zwei Spaltentitel klicken und bei gedrückter Maustaste ziehen.

2. Drücken Sie die linke Maustaste, und halten Sie sie gedrückt.
3. Bewegen Sie die Maus, um die neue Spaltenbreite festzulegen.
4. Lassen Sie die linke Maustaste los, wenn Sie die gewünschte Spaltenbreite erreicht haben.

Alternativ können Sie mit der rechten Maustaste auf den Feldnamen klicken und im Kontextmenü **FELDBREITE** auswählen. Tippen Sie anschließend die Felddbreite (ungefähre Anzahl der Zeichen; wie in Excel) ein. Per Doppelklick zwischen zwei Spaltentitel können Sie auf die Standardbreite zurücksetzen.

2.5.4 Felder sortieren

Die Reihenfolge der Felder – also die Anordnung der Spalten, nicht die Sortierung der Datensätze – können Sie auf zwei Wegen verändern:


1. in der Entwurfsansicht (empfohlen; siehe Abschnitt 2.4.2, »Verschieben von Feldern«)
2. in der Datenblattansicht per Drag & Drop auf den Feldnamen (Titel der Spalte)

Über den zweiten Weg wird die physikalische Anordnung der Felder in der Tabelle *nicht* verändert. Sie erkennen dies daran, dass die Anordnung der Felder in der Entwurfsansicht oder im Datenbankdiagramm unverändert bleibt.

2.5.5 Felder ein- und ausblenden

Sie können Felder in der Datenblattansicht nicht nur neu sortieren, sondern auch ausblenden. Ausgeblendete Felder werden *nicht* gelöscht, und genauso wie beim Sortieren der Felder wird die physikalische Anordnung der Felder in der Tabelle *nicht* verändert.

So blenden Sie ein Feld aus:

1. Markieren Sie das Feld, indem Sie auf den Feldnamen (Titel der Spalte) klicken.
2. Optional können Sie mit gedrückter -Taste mehrere Felder auswählen.
3. Klicken Sie mit der *rechten* Maustaste auf den markierten Spaltentitel.
4. Wählen Sie im Kontextmenü FELDER AUSBLENDEN.

Und so blenden Sie ausgeblendete Felder wieder ein:

1. Klicken Sie auf einem beliebigen Spaltentitel die *rechte* Maustaste.
2. Wählen Sie im Kontextmenü FELDER WIEDER EINBLENDEN aus.
3. Das Dialogfeld SPALTEN EINBLENDEN wird angezeigt:

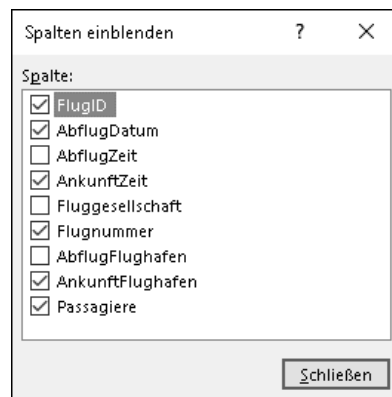


Abbildung 2.45 Im Dialogfeld »Spalten einblenden« können Sie per Haken die Felder ein- oder ausblenden.

4. Setzen Sie einen Haken vor alle Felder, die wieder eingeblendet werden sollen.
5. Klicken Sie auf SCHLIESSEN.

Ausgeblendete Felder werden leicht übersehen

Wenn Sie nur an bestimmten Feldern einer Tabelle interessiert sind, ist eine Abfrage der beste Weg (mehr dazu in Abschnitt 3.1.1, »Vertikales Filtern – Felder auswählen«). Mit ausgeblendeten Feldern erreichen Sie das gleiche Ziel.

Mir passiert es immer wieder, dass ich ausgeblendete Felder übersehe. Letztendlich bringt ein Blick in die Entwurfsansicht der Tabelle Klarheit, denn hier kann man nichts ausblenden oder verstecken!


In Bezug auf Tabellen bin ich ziemlicher Purist: Ich nutze die Datenblattansicht nur zur Datenerfassung. Die Anordnung von Feldern, vertikales und horizontales Filtern sowie das Sortieren von Datensätzen erledige ich gerne mit Hilfe von Abfragen.

2.5.6 Felder fixieren

Eine Tabelle mit besonders vielen Feldern kann unter Umständen nicht vollständig am Bildschirm angezeigt werden; ich habe aber auch schon von Enthusiasten gehört, die eine ganze Batterie von Monitoren nebeneinander aufbauen, um *alle* Spalten im Blick zu haben.

Gehören auch Sie zu den Leuten, die lieber auf Nackenschmerzen verzichten möchten? Dann können Sie die horizontale Bildlaufleiste am unteren Rand der Datenblattansicht verwenden, um die Anzeige der Felder zu verschieben.

Besonders wichtige Felder wie beispielsweise den Primär- und den Alternativschlüssel können Sie im linken Bereich der Datenblattansicht fixieren.

1. Markieren Sie das Feld, indem Sie auf den Feldnamen (Titel der Spalte) klicken.
2. Optional können Sie mit gedrückter -Taste mehrere Felder auswählen.
3. Klicken Sie mit der *rechten* Maustaste auf den markierten Spaltentitel.
4. Wählen Sie im Kontextmenü FELDER FIXIEREN aus.

Fixierte Felder werden immer links angezeigt

In der Datenblattansicht werden die fixierten Felder immer im linken Bereich dargestellt. Getrennt davon durch eine schmale, vertikale Trennlinie befinden sich die nicht fixierten Felder immer auf der rechten Seite. Die horizontale Bildlaufleiste wirkt sich immer nur auf den rechten Bereich mit den nicht fixierten Feldern aus.

Und so heben Sie die Fixierung von Feldern wieder auf:

1. Klicken Sie auf einem beliebigen Spaltentitel die *rechte* Maustaste.
2. Wählen Sie im Kontextmenü **FIXIERUNG ALLER FELDER AUFHEBEN** aus.

Es ist leider nicht möglich, nur ein einzelnes Feld aus der Fixierung herauszunehmen. Nachdem Sie die Fixierung aufgehoben haben, sind die Felder unter Umständen anders angeordnet als vorher. Wie in Abschnitt 2.5.4, »Felder sortieren«, gezeigt, können Sie die ursprüngliche Anordnung der Felder wiederherstellen.

2.5.7 Datensätze sortieren

Nachdem wir uns in den letzten Abschnitten mit der Anordnung von Feldern beschäftigt haben, schauen wir uns jetzt an, wie Sie Datensätze sortieren. Eigentlich ist Sortierung ein Thema für Abfragen. In Access können Sie aber auch auf Tabellenebene sortieren. Zunächst möchte ich Ihnen einen Merksatz mitgeben, der Ihnen das Verständnis von Sortierung in Datenbanken einfacher machen kann:

Die Reihenfolge der Datensätze in einer Tabelle kann sich jederzeit ändern!

Verlassen Sie sich *niemals* darauf, dass die Anordnung von Datensätzen in einer Tabelle unverändert bleibt. Sie sind es vielleicht von Excel gewohnt, dass die Reihenfolge der Datensätze in einer Tabelle unverändert bleibt, solange Sie nicht neu sortieren. Für Datenbanken gilt das jedoch nicht!

Wer es genau wissen will: Die standardmäßige Sortierung in Datenbanken

Der eine oder andere Access-Benutzer wird mir entgegenen, dass meine Aussage so nicht stimmt. Zugegeben, in den meisten Fällen bleibt die Reihenfolge der Datensätze so, wie sie eingegeben wurden (um genau zu sein: sortiert nach dem Primärschlüssel). Zumindest gilt das für Access und solange Sie nur eine Tabelle betrachten.

Anders sieht es jedoch mit Abfragen und Server-Datenbanken aus. Mit dem Merksatz »Die Reihenfolge der Datensätze kann sich jederzeit ändern!« sind Sie immer auf der sicheren Seite und können sich dadurch eine ganze Reihe unschöner Überraschungen ersparen.

Wenn die Sortierung standardmäßig nicht festgelegt ist, dann heißt das genau eines: Sie müssen die Sortierung der Datensätze selbst festlegen, wenn Sie dies wünschen. Andernfalls rechnen Sie bitte damit, dass die Datensätze wild durcheinandergewürfelt angezeigt werden.

Datensätze in der Datenblattansicht sortieren

In Access können Sie die Sortierung auf Tabellenebene sowohl in der Datenblattansicht als auch in der Entwurfsansicht einstellen. So sortieren Sie in der Datenblattansicht die Tabelle *tblFlug* nach »AbflugDatum«, »AnkunftZeit« und »Flugnummer:«

1. Öffnen Sie die Datenbank *O2_Access_als_Datenbank_Tabellen\2.3.12_Fluege.accdb* aus den Materialien zum Buch.
2. Öffnen Sie die Tabelle *tblFlug* in der Datenblattansicht.

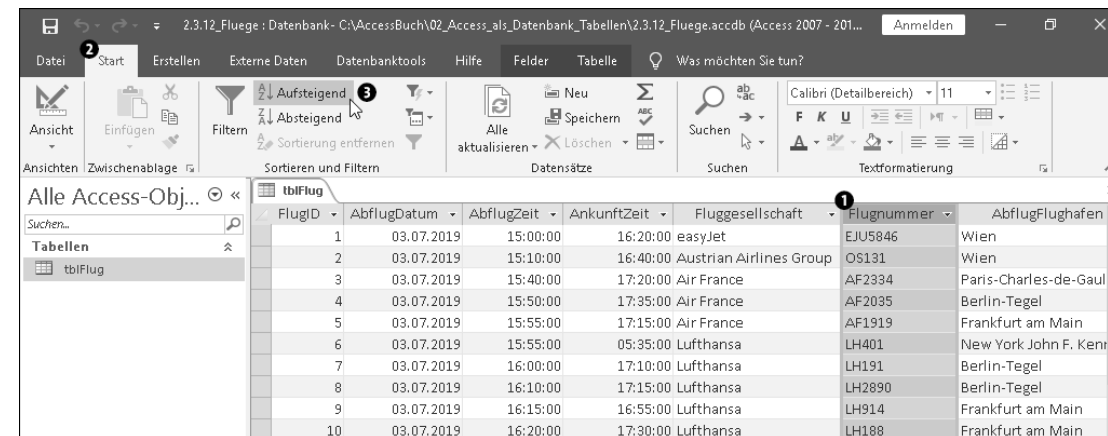


Abbildung 2.46 In der Datenblattansicht können Sie Datensätze auf- oder absteigend sortieren.

3. Markieren Sie das Feld »Flugnummer«, indem Sie auf den Feldnamen (Titel der Spalte) klicken ❶.
4. Gehen Sie zu **START** ❷ • **SORTIEREN UND FILTERN**.
5. Klicken Sie auf **AUFSTIEGEND** ❸.
Die Tabelle wird dadurch nach »Flugnummer« aufsteigend sortiert. Sie erkennen dies an einem kleinen Pfeil neben dem Feldnamen.
6. Wiederholen Sie die Schritte 3 bis 5 für das Feld »AnkunftZeit«.
Die Sortierung nach »Flugnummer« geht dadurch nicht verloren – wenn Sie genau hinschauen, erkennen Sie neben den Feldnamen sowohl für das Feld »AnkunftZeit« als auch für »Flugnummer« einen kleinen Pfeil.
7. Wiederholen Sie die Schritte 3 bis 5 für das Feld »AbflugDatum«.

Nach meinem Empfinden etwas gewöhnungsbedürftig ist, dass die Sortierung umgekehrt eingetragen werden muss. Obwohl erstens nach »AbflugDatum«, zweitens nach »AnkunftZeit« und drittens nach »Flugnummer« sortiert wird, müssen Sie eine Eingabe genau umgekehrt vornehmen, also erst »Flugnummer« auswählen und sortieren, dann »AnkunftZeit« und schließlich »AbflugDatum«.

Das Ergebnis finden Sie in den Materialien zum Buch in der Datenbank *O2_Access_als_Datenbank_Tabellen\2.5.7_Fluege_Sortierung.accdb*.

Die Sortierung von Datensätzen wieder entfernen

So entfernen Sie die Sortierung von Datensätzen wieder:

1. Öffnen Sie die Datenbank *O2_Access_als_Datenbank_Tabellen\2.5.7_Fluege_Sortierung.accdb* aus den Materialien zum Buch.
2. Öffnen Sie die Tabelle *tblFlug* in der Datenblattansicht.
3. Gehen Sie zu **START • SORTIEREN UND FILTERN**.
4. Klicken Sie auf **SORTIERUNG ENTFERNEN**.

Üblicherweise wird Access die Datensätze jetzt wieder nach dem Primärschlüssel, also nach »FlugID«, aufsteigend sortieren. Aber wie gesagt, meine Hand lege ich dafür nicht ins Feuer.

Datensätze in der Entwurfsansicht sortieren

Etwas logischer als in der Datenblattansicht empfinde ich die Sortierung von Datensätzen über die Entwurfsansicht (Abbildung 2.47).

1. Öffnen Sie die Datenbank *O2_Access_als_Datenbank_Tabellen\2.3.12_Fluege.accdb* in den Materialien zum Buch.
2. Öffnen Sie die Tabelle *tblFlug* in der Entwurfsansicht.

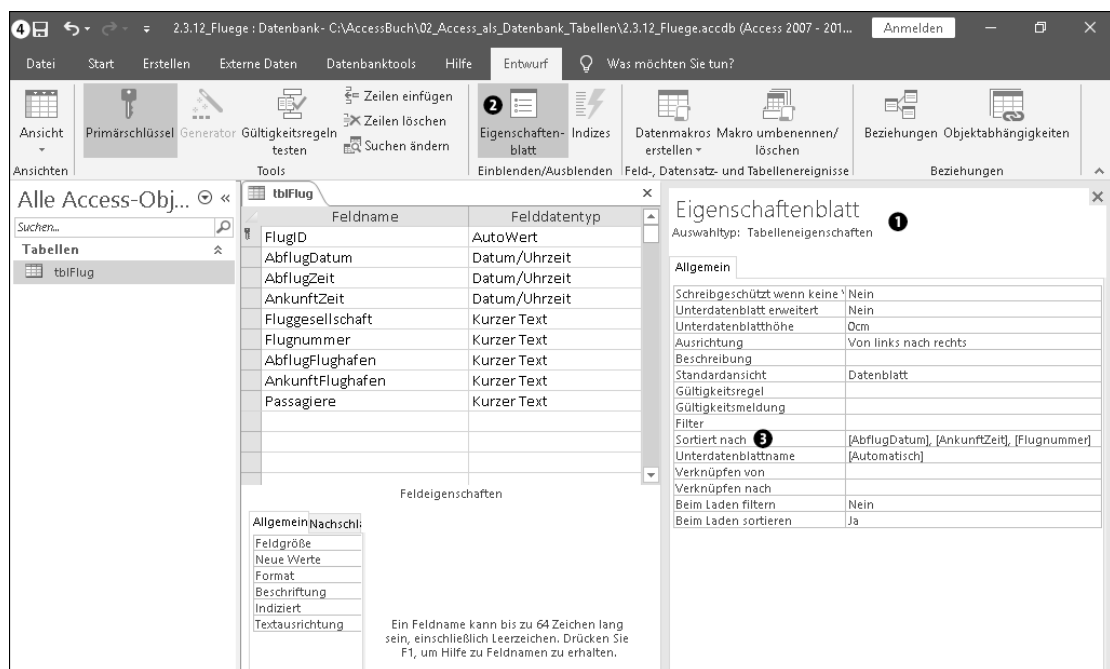


Abbildung 2.47 In der Entwurfsansicht können Sie die Sortierung in den Tabelleneigenschaften festlegen.

3. Falls die Tabelleneigenschaften **1** noch nicht sichtbar sind: Klicken Sie auf **ENTWURF • EINBLENDEN/AUSBLENDEN • EIGENSCHAFTENBLATT 2**.
4. Tragen Sie im **EIGENSCHAFTENBLATT** (Tabelleneigenschaften) **1** unter **SORTIERT NACH 3** [AbflugDatum], [AnkunftZeit], [Flugnummer] ein.
5. Klicken Sie in der Symbolleiste für den Schnellzugriff auf **SPEICHERN 4** (oder **Strg** + **S**).

Tragen Sie die Feldnamen in eckigen Klammern ([und]) und getrennt durch Kommata ein, und zwar genau in der Reihenfolge der Sortierebenen: erst »AbflugDatum«, dann »AnkunftZeit« und schließlich »Flugnummer«.

Wenn Sie die Sortierung in der Datenblattansicht vornehmen, erstellt Access den Eintrag unter **SORTIERT NACH** automatisch. Lassen Sie sich bitte nicht dadurch verwirren, dass Access vor jedem Feldnamen noch den Tabellennamen vermerkt hat:

[tblFlug].[AbflugDatum], [tblFlug].[AnkunftZeit], [tblFlug].[Flugnummer]

Diese *voll qualifizierte Angabe* der Felder ist eigentlich nicht notwendig, da nur Daten aus einer einzigen Tabelle sortiert werden. Es schadet aber auch nicht.

Die Sortierung anwenden


Wenn Sie in den Tabelleneigenschaften unter **BEIM LADEN SORTIEREN** die Option **JA** auswählen, werden die Datensätze sortiert, sobald Sie die Tabelle öffnen. Wenn Sie hier **NEIN** eintragen, wird die Form der Einstellung zur Sortierung zwar gespeichert, aber noch nicht angewendet. Klicken Sie in diesem Fall auf **START • SORTIEREN UND FILTERN • ERWEITERTE FILTEROPTIONEN • FILTER/SORTIERUNG ANWENDEN**, um die Datensätze tatsächlich sortieren zu lassen.

Zusammenfassung

Letztendlich führen beide Wege (Sortierung in der Datenblattansicht oder in der Entwurfsansicht) zum gleichen Ergebnis, nämlich zu einer Formel in der Tabelleneigenschaft **SORTIERT NACH**. Bitte vergessen Sie nicht, dass die Sortiereinstellung (die Formel) dauerhaft abgespeichert wird, wenn Sie auf **SPEICHERN** klicken (oder **Strg** + **S** drücken).

2.5.8 Suchfunktionen

Genauso wichtig wie das Sortieren ist das Suchen von Datensätzen in einer Datenbank. Wenn Sie beispielsweise die Flugnummer LH401 in unserer Tabelle *tblFlug* suchen möchten, können Sie den Suchtext unten in das Suchfeld neben den Navigationsschaltflächen einge-

ben. Access springt automatisch zum ersten passenden Datensatz. Mit der -Taste springen Sie zum nächsten passenden Datensatz.

Neben dem Suchfeld gibt es in Access den Dialog SUCHEN UND ERSETZEN, den Sie über **START • SUCHEN • SUCHEN** (oder **Strg + F**) aufrufen:

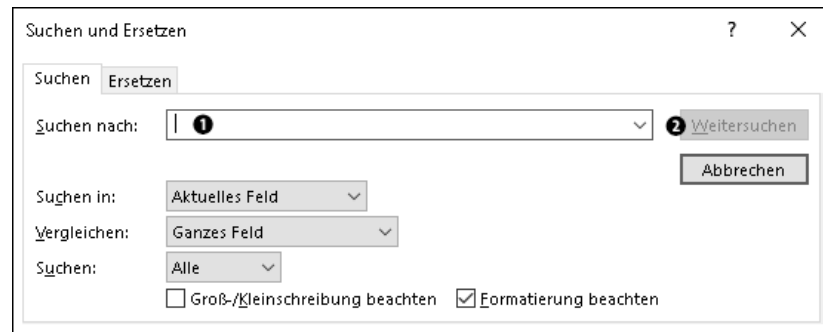


Abbildung 2.48 Der Dialog »Suchen und Ersetzen« funktioniert ähnlich wie in den anderen Office-Programmen.

Die Suche mit dem Dialog funktioniert genauso, wie Sie das vielleicht schon von den anderen Office-Programmen her kennen (Abbildung 2.48):

1. Tragen Sie den Suchtext ein **1**.
2. Klicken Sie auf WEITERSUCHEN **2**.

Access zeigt daraufhin den ersten passenden Datensatz an. Wenn Sie noch einmal auf WEITERSUCHEN klicken, gelangen Sie zum nächsten passenden Datensatz. Es gibt ein paar Optionen, mit denen Sie die Suche verfeinern können:

1. SUCHEN IN

Wählen Sie **AKTUELLES FELD**, wenn Sie nur in einem bestimmten Feld suchen möchten. Die Option **AKTUELLES DOKUMENT** führt dazu, dass in den Inhalten aller Felder gesucht wird.

2. VERGLEICHEN

Hier können Sie festlegen, ob der Suchtext mit dem vollständigen Feldinhalt übereinstimmen soll (Option: **GANZES FELD**). Zum Beispiel exakt der Flug LH401, nicht aber LH4011.

Verwenden Sie die Option **ANFANG DES FELDHALTES**, wenn Sie Feldinhalte finden möchten, die mit einer bestimmten Zeichenfolge beginnen, beispielsweise den Suchtext »LH« für alle Lufthansa-Flüge.

TEIL DES FELDHALTES findet diejenigen Datensätze, in denen der Suchtext irgendwo im Feldinhalt auftaucht. Die Suche nach »401« findet dann sowohl LH401 und 401abc als auch LH4011.

3. SUCHEN: ALLE, AUFWÄRTS oder ABWÄRTS

Hier legen Sie fest, in welche Richtung gesucht werden soll (vom aktuellen Datensatz aus gesehen).

4. GROSS-/KLEINSCHREIBUNG BEACHTEN

Wenn Sie hier ein Häkchen setzen, wird die Groß-/Kleinschreibung bei der Suche berücksichtigt. »LH401« ist dann etwas anderes als »Lh401«.

5. FORMATIERUNG BEACHTEN

In Abschnitt 2.6, »Formatierungen in Tabellen«, werde ich Ihnen zeigen, dass Feldinhalte bei aktivierter Formatierung anders angezeigt werden, als sie abgespeichert sind. Mit dem Häkchen bei **FORMATIERUNG BEACHTEN** durchsucht Access die Feldinhalte so, wie sie am Bildschirm zu sehen sind. Ohne das Häkchen werden nur die abgespeicherten Feldinhalte ohne Formatierung durchsucht. Solange Sie keine Formatierung verwenden, spielt diese Option keine Rolle.

Über das Registerblatt **ERSETZEN** (oder **Strg + H**) können Sie einen bestimmten Suchtext durch einen anderen Text ersetzen, und zwar in einer ähnlichen Form, wie Sie es von Word oder Excel her gewohnt sind (Abbildung 2.49):

1. Tragen Sie den Suchtext ein **1**.
2. Geben Sie den neuen Text in das Feld **ERSETZEN DURCH** **2** ein.
3. Klicken Sie auf **WEITERSUCHEN** **3**, um den ersten Treffer zu finden.
4. Klicken Sie auf **ERSETZEN** **4**, um den gefundenen Text zu ersetzen und zum nächsten Treffer zu springen.
5. Klicken Sie entweder auf **WEITERSUCHEN** **3** oder auf **ERSETZEN** **4**, um einen Treffer zu überspringen bzw. zu ersetzen. Wenn Sie alle Treffer auf einmal ersetzen möchten, geht es mit einem Klick auf **ALLE ERSETZEN** **5** schneller.

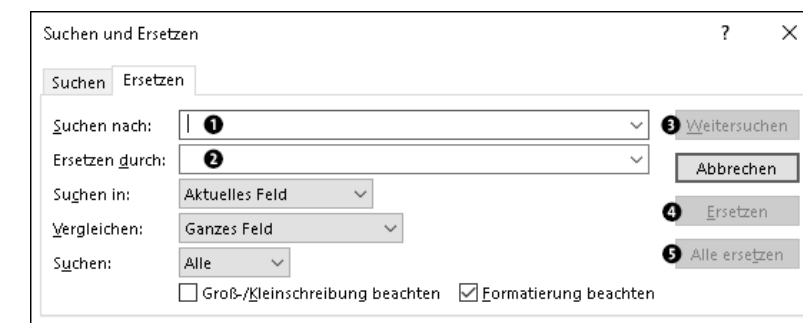


Abbildung 2.49 Die Funktion »Ersetzen« eignet sich eher für überschaubare Datenänderungen. Für Datenanpassungen im großen Stil sind Aktualisierungsabfragen das Mittel der Wahl.

Beim Ersetzen kein Undo!

Leider gibt es die Funktion RÜCKGÄNGIG für ALLE ERSETZEN nicht. Sie müssen daher das Ersetzen aller Treffer ausdrücklich bestätigen.

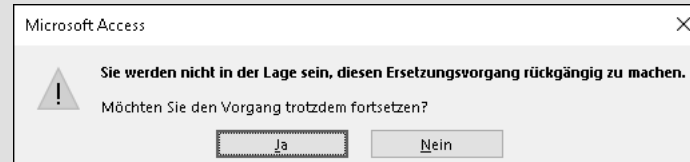


Abbildung 2.50 Der Vorgang »Alle ersetzen« lässt sich leider nicht rückgängig machen – Sie müssen das Ersetzen daher ausdrücklich bestätigen.

Die Funktion SUCHEN UND ERSETZEN ist ein sehr mächtiges Werkzeug, mit dem Sie mehrere Datensätze gleichzeitig bearbeiten können. Gleichwohl sind Abfragen noch besser für diese Aufgabe geeignet. Mit Abfragen können Sie komplexere Suchanfragen durchführen und das Ersetzen noch genauer steuern. In Access kommen dafür zwei Typen von Abfragen in Frage:

1. *Auswahlabfragen* zum Suchen von Daten (Abschnitt 3.1.2, »Horizontales Filtern – Datensätze auswählen«)
2. *Aktualisierungsabfragen* zum Ändern einer großen Zahl von Datensätzen (Abschnitt 3.5.3, »Aktualisierungsabfragen«).

Diese beiden Abfragetypen stellen die optimalen Werkzeuge für Datenbanken dar und haben auch mit großen Datenmengen keine Probleme. Der Dialog SUCHEN UND ERSETZEN in der Datenblattansicht ist eher der kleine Bruder für die schnelle Suche mal eben zwischendurch.

2.5.9 Datensätze filtern

In Access gibt es zwei Konzepte, mit denen Sie Daten in einer Tabelle wiederfinden: entweder *Suchen* oder *Filtern*.

Suchen	Filtern
nur in der Datenblattansicht verfügbar	entweder über Abfragen oder in der Datenblattansicht
von einem Treffer zum nächsten springen	Nur die Treffer werden angezeigt.
Suchen und Ersetzen ist möglich.	Das Ersetzen von Feldinhalten ist mit dem Filter direkt nicht möglich (für diese Aufgabe gibt es Aktualisierungsabfragen).

Tabelle 2.21 In Access gibt es zwei Konzepte, wie Sie Daten in einer Tabelle wiederfinden

Während es im letzten Abschnitt um das Konzept *Suchen* ging, stelle ich Ihnen in diesem Abschnitt das Konzept *Filtern in der Datenblattansicht* vor. Der wesentliche Unterschied ist, dass beim Filtern nur die Treffer angezeigt werden; alle anderen Datensätze werden ausgeblendet.

Mit dem Konzept des Filterns behalten Sie auch bei großen Datenmengen den Überblick

Der Vorteil des Filterns ist es, dass Sie nur das sehen, was Sie wirklich interessiert: die Treffer. Dies ist bei einer großen Anzahl von Datensätzen der einzig praktikable Weg, den Überblick über alle Treffer zu behalten, speziell dann, wenn Sie mehrere Filter hintereinanderschalten möchten. Hier ist ein Beispiel von drei Filtern, die hintereinandergeschaltet sind: »Alle Lufthansa-Flüge, die vor 18:00 Uhr in Berlin-Tegel starten.« Solche komplexen Suchanfragen sind typisch für Datenbanken und lassen sich in Access nur über das Konzept des Filterns realisieren. Im Übrigen basieren Auswahlabfragen auch auf diesem Konzept.

Access bietet eine ganze Reihe von Filtern in der Datenblattansicht an, die ich Ihnen nun vorstellen werde. Sie werden sehen, dass letztendlich jeder Filter nichts anderes als eine Formel ist.

Feldinhalte auswählen

Im Spaltenkopf können Sie über den kleinen Dropdown-Pfeil das Spaltenmenü aufrufen. Hier wählen Sie diejenigen Einträge aus, die Sie filtern möchten.

1. Öffnen Sie die Datenbank *O2_Access_als_Datenbank_Tabellen\2.3.12_Fluege.accdb* aus den Materialien zum Buch.
2. Öffnen Sie die Tabelle *tblFlug* in der Datenblattansicht (Abbildung 2.51).



Abbildung 2.51 Im Spaltenmenü wählen Sie die Feldinhalte aus, die Sie filtern möchten.

3. Klicken Sie im Spaltentitel von »Fluggesellschaft« auf den Dropdown-Pfeil ❶.
4. Entfernen Sie das Häkchen bei (ALLE AUSWÄHLEN) ❷. Damit entfernen Sie die Häkchen von allen Einträgen.
5. Setzen Sie das Häkchen bei AIR FRANCE ❸ und bei BRITISH AIRWAYS ❹.
6. Klicken Sie auf OK ❺.

Auf diese Weise filtern Sie alle Flüge von Air France oder British Airways aus der Gesamtmenge aller Flüge.

NULL-Werte filtern

Um Datensätze zu filtern, in denen der Wert NULL steht, wählen Sie (LEER) aus.

Nun können Sie ebenso andere Felder filtern und somit einen Filter mit mehreren Stufen erzeugen, beispielsweise ein weiterer Filter im Feld »AbflugFlughafen« auf »Berlin-Tegel«, um alle Flüge von Air France und British Airways mit Abflug in Berlin-Tegel zu filtern.

Einen Auswahlfiler einrichten

Ein Auswahlfiler basiert immer auf dem aktuell gewählten Feldinhalt. Wenn Ihr aktueller Datensatz beispielsweise ein Air-France-Flug ist, können Sie sehr schnell einen Auswahlfiler einrichten, um alle Air-France-Flüge zu filtern.

1. Öffnen Sie die Datenbank *O2_Access_als_Datenbank_Tabellen\2.3.12_Fluege.accdb* aus den Materialien zum Buch.
2. Öffnen Sie die Tabelle *tblFlug* in der Datenblattansicht.
3. Gehen Sie zum zweiten Datensatz (»FlugID« = 2) und dort in das Feld »Fluggesellschaft« ❶.

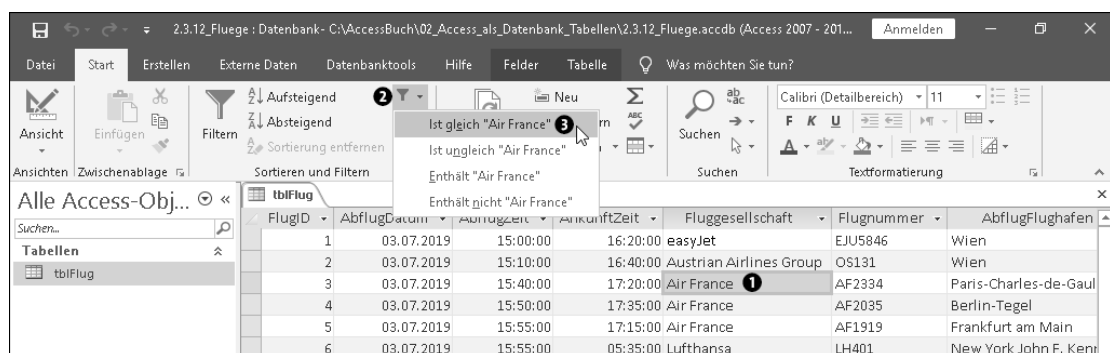


Abbildung 2.52 Auf dem aktuellen Feldinhalt basierend können Sie einen Auswahlfiler erstellen, um ähnliche Datensätze zu finden.

4. Gehen Sie zu START • SORTIEREN UND FILTERN • AUSWAHL ❷.
5. Wählen Sie IST GLEICH »AIR FRANCE« ❸ aus.

Ebenso können Sie den Auswahlfiler im Feld über das Kontextmenü (rechte Maustaste) einstellen. Neben IST GLEICH gibt es andere Filteroperatoren, die sich je nach Felddatentyp unterscheiden.

Textfilter, Zahlenfilter und Datumsfilter

Wenn Ihnen die Filteroperatoren der Auswahlfiler nicht ausreichen, können Sie aus einer Liste weiterer Operatoren wählen. Die verfügbaren Filter unterscheiden sich je nach Felddatentyp des gewählten Feldes:

Filter	Operatoren
TEXTFILTER	GLEICH
	NICHT GLEICH
	BEGINNT MIT
	BEGINNT NICHT MIT
	ENTHÄLT
	ENTHÄLT NICHT
	ENDET MIT
	ENDET NICHT MIT
ZAHLENFILTER	GLEICH
	NICHT GLEICH
	KLEINER ALS
	GRÖßER ALS
	ZWISCHEN
DATUMSFILTER	GLEICH
	NICHT GLEICH
	VOR
	NACH
	ZWISCHEN
	MORGEN

Tabelle 2.22 Je nach Felddatentyp angepasste Filter bieten Ihnen einen ganze Fülle von Filteroperatoren.

Filter	Operatoren
DATUMSFILTER (Forts.)	HEUTE
	GESTERN
	NÄCHSTE WOCH
	DIESE WOCH
	LETZTE WOCH
	NÄCHSTER MONAT
	DIESER MONAT
	LETZTER MONAT
	NÄCHSTES QUARTAL
	DIESES QUARTAL
	LETZTES QUARTAL
	NÄCHSTES JAHR
	DIESES JAHR
	BIS ZUM AKTUELLEN DATUM
	LETZTES JAHR
	VERGANGENHEIT
	ZUKUNFT
	ALLE DATEN IN DIESEM ZEITRAUM • QUARTAL 1
	...
	ALLE DATEN IN DIESEM ZEITRAUM • QUARTAL 4
ALLE DATEN IN DIESEM ZEITRAUM • JANUAR	
...	
ALLE DATEN IN DIESEM ZEITRAUM • DEZEMBER	

Tabelle 2.22 Je nach Felddatentyp angepasste Filter bieten Ihnen einen ganzen Fülle von Filteroperatoren. (Forts.)

Und so gehen Sie vor, um einen Text-, Zahlen- oder Datumsfilter einzurichten:

1. Wählen Sie das Feld aus, auf das sich der Filter beziehen soll.
2. Klicken Sie im Spaltentitel auf den Dropdown-Pfeil. Je nach Felddatentyp wird im Spaltenmenü entweder TEXTFILTER, ZAHLENFILTER oder DATUMSFILTER angezeigt.
3. Wählen Sie im Untermenü den passenden Filteroperator aus.
4. Bei einigen Filteroperatoren müssen Sie anschließend noch einen Wert eingeben.

Ich muss ganz ehrlich sagen: Diese Fülle an Filteroperatoren reicht aus, um die allermeisten Suchvorgänge zu erledigen. Und richtig angenehm empfinde ich die schnelle Bedienung!

Erweiterte Filteroptionen

Unter **START • SORTIEREN UND FILTERN • ERWEITERT** finden Sie den *formularbasierten Filter*. Im Prinzip ist dieser Filter eine besondere Form, nach Feldinhalten zu filtern. Beim Aktivieren des formularbasierten Filters schaltet Access von der Datenblattansicht in ein primitives Formular um.

In diesem Formular können Sie über Dropdown-Felder die gewünschten Feldinhalte auswählen. Nachdem Sie auf **START • SORTIEREN UND FILTERN • FILTER EIN/AUS** geklickt haben, gelangen Sie zurück zur gefilterten Datenblattansicht der Tabelle.



Abbildung 2.53 Über Dropdown-Felder können Sie in einem formularbasierten Filter die gewünschten Werte auswählen.

Schließlich gibt es noch die Option **SPEZIALFILTER/-SORTIERUNG** mit weiteren Funktionen (**AUS ABFRAGE LADEN ...** sowie **ALS ABFRAGE SPEICHERN ...**). Wenn Sie diesen Filter wählen, gelangen Sie zu einer Ansicht, die fast identisch mit der Entwurfsansicht einer Abfrage ist. In Kapitel 3, »Daten filtern, sortieren und zusammenfassen: Abfragen«, werde ich Ihnen Abfragen und die Entwurfsansicht von Abfragen ausführlich vorstellen. An dieser Stelle möchte ich daher nicht weiter vorgreifen. Nur so viel: Wenn Sie einen Spezialfilter oder eine Spezialsortierung erstellen wollen, dann können Sie stattdessen auch gleich eine *Auswahlabfrage* erstellen.

Das Filtern von Datensätzen ein- und ausschalten

Das Konzept des Filterns hat natürlich auch eine Tücke: Auf den ersten Blick übersieht man leicht, dass nicht alle Datensätze einer Tabelle angezeigt werden, sondern nur die Treffer. Daher ist es sehr wichtig, dass Sie sich immer darüber im Klaren sind, ob Sie gerade alle Datensätze sehen oder ob ein Filter aktiv ist. An mehreren Stellen können Sie in Access erkennen, ob ein Filter aktiv ist (Abbildung 2.54):

- ▶ Die Schaltfläche **START • SORTIEREN UND FILTERN • FILTER EIN/AUS** ❶ ist dunkel hinterlegt.
- ▶ Im Spaltentitel von »Fluggesellschaft« erscheint ein kleines Filtersymbol ❷.
- ▶ Unten erscheint rechts neben den Navigationsschaltflächen die Schaltfläche **GEFILTERT** ❸. Diese Schaltfläche ist zudem orange hinterlegt.
- ▶ In der Statusleiste erscheint **FILTERED** ❹.

Ebenso gibt es in Access mehrere Möglichkeiten, einen Filter einzuschalten und wieder auszuschalten. Am einfachsten sind nach meinem Empfinden die beiden Schaltflächen, die bei aktiviertem Filter farblich markiert werden (**START • SORTIEREN UND FILTERN • FILTER EIN/AUS** ❶ sowie die Schaltfläche **GEFILTERT** ❸ unten rechts neben den Navigationsschaltflächen). Über diese Schaltflächen können Sie den Filter ein- und ausschalten.

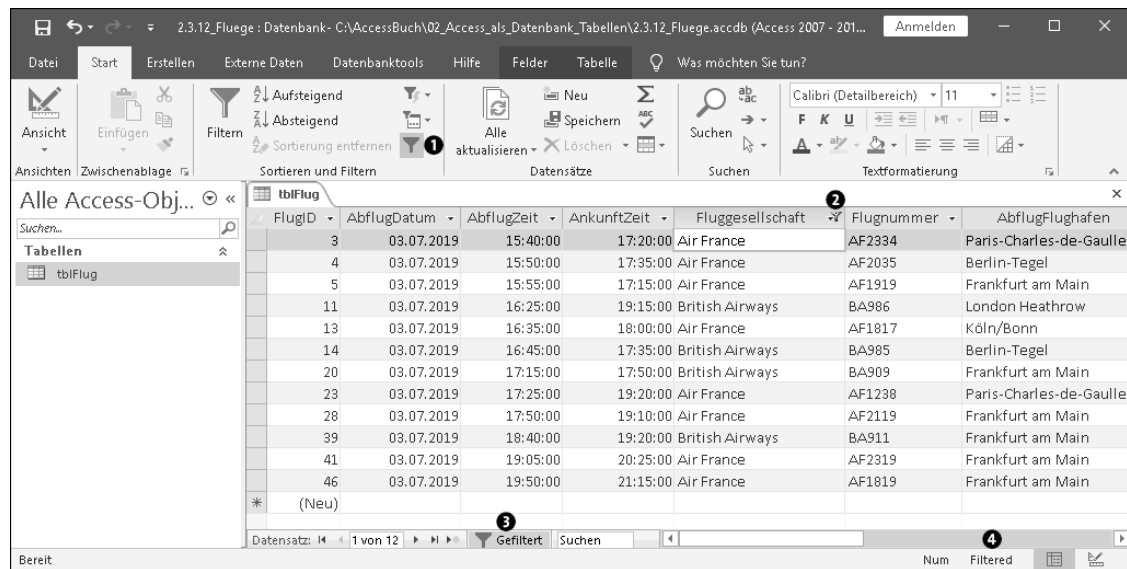


Abbildung 2.54 Wichtig ist, dass Sie sich immer darüber im Klaren sind, ob alle Datensätze einer Tabelle angezeigt werden oder ob ein Filter aktiv ist.

Den Filter wieder entfernen

Wenn Sie einen Filter **ausschalten**, werden wieder alle Datensätze der Tabelle angezeigt. Die Einstellungen des Filters bleiben jedoch erhalten! Über **SPEICHERN** (oder **Strg** + **S**) wer-

den die Filtereinstellungen sogar dauerhaft abgespeichert. Wenn Sie später einen Filter wieder einschalten, dann werden die vorhandenen Einstellungen des Filters erneut angewendet, und Sie erhalten nur die Datensätze der Treffer.

Etwas anderes ist es, einen Filter zu **löschen**. Beim Löschen eines Filters werden die Filtereinstellungen wirklich gelöscht. Wenn Sie dann auf **SPEICHERN** klicken (oder **Strg** + **S** drücken), werden die Filtereinstellungen auch dauerhaft entfernt.

- ▶ **Filter für ein Feld löschen:**
Klicken Sie im Spaltenmenü auf **FILTER LÖSCHEN AUS <SPALTENNAME>**. Damit wird nur der Filter für dieses eine Feld gelöscht.
- ▶ **Den gesamten Filter löschen:**
Klicken Sie auf **START • SORTIEREN UND FILTERN • ERWEITERT • ALLE FILTER LÖSCHEN**, um die Filter aller Felder zu löschen.

Datensätze in der Entwurfsansicht filtern

Wir sollten uns einmal die Filtereinstellungen genauer ansehen. Immer dann, wenn Sie auf einem der oben beschriebenen Wege einen Filter einrichten, ändert Access für Sie genau eine Einstellung: die Tabelleneigenschaft **FILTER**. Sie finden diese Einstellung in der Entwurfsansicht:

1. Öffnen Sie die Datenbank *O2_Access_als_Datenbank_Tabellen\2.3.12_Fluege.accdb* aus den Materialien zum Buch.
2. Öffnen Sie die Tabelle *tblFlug* in der Datenblattansicht.
3. Legen Sie einen beliebigen Filter wie oben beschrieben fest (z. B. nur die Flüge von Air France).
4. Schalten Sie in die Entwurfsansicht um (Abbildung 2.55).
5. Falls die Tabelleneigenschaften ❶ noch nicht sichtbar sind: Klicken Sie auf **ENTWURF • EINBLENDEN/AUSBLENDEN • EIGENSCHAFTENBLATT** ❷.
6. Unter **FILTER** ❸ hat Access die Filtereinstellung eingetragen:
`([tblFlug].[Fluggesellschaft]="Air France")`

Jede Filtereinstellung ist also nichts anderes als eine Formel. Die Syntax ist identisch mit der einer Tabellengültigkeitsregel (vergleiche Abschnitt 2.3.13, »Tabelleneinschränkungen«). In unserem Fall hat Access eine Formel mit voll qualifizierter Angabe des Feldnamens (d. h. mit Angabe des Tabellennamens) erzeugt, obwohl das bei einem Filter auf einer einzigen Tabelle eigentlich nicht notwendig ist. Ebenso können Sie sich die runden Klammern sparen:

```
[Fluggesellschaft]="Air France"
```

In der Entwurfsansicht unter **TABELLENEIGENSCHAFTEN • FILTER** können Sie die Formel für die Filtereinstellungen auch direkt eintragen.

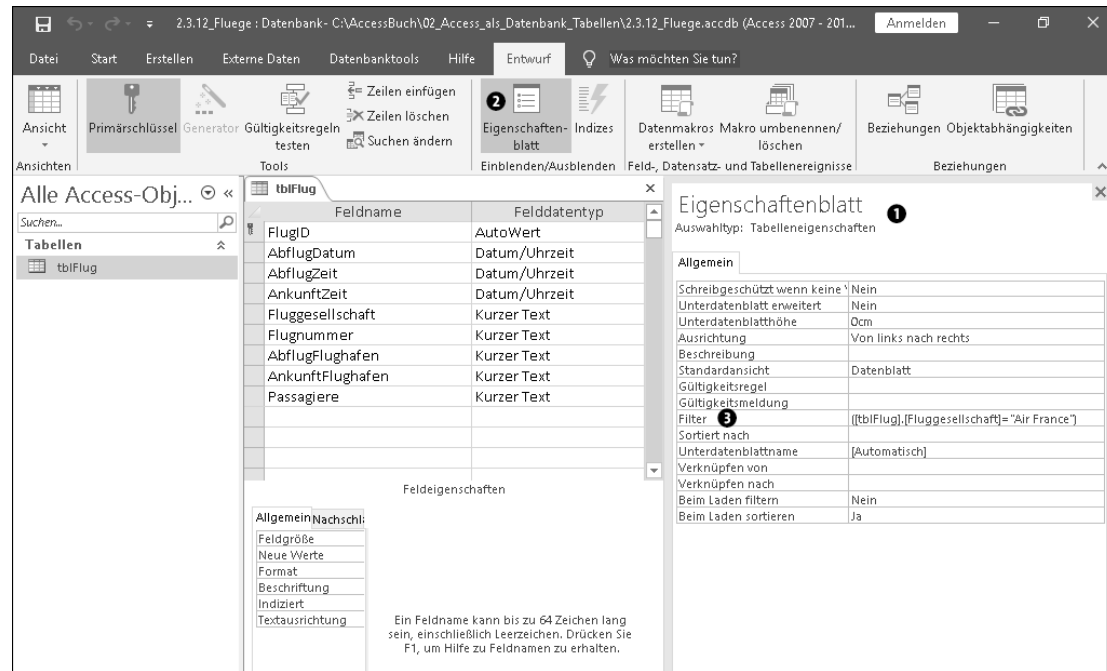


Abbildung 2.55 In der Tabelleneigenschaft »Filter« wird die Filtereinstellung als Formel eingetragen.

Mit Hilfe der Filtereinstellungen können Sie interessante Formeln erlernen

Jeder Filter ist letztendlich eine Formel, egal, ob Sie die Formel direkt eingetragen haben oder ob sie von Access generiert wurde. Das Schöne ist, dass Sie an dieser Stelle sehr viel über Formeln lernen können: Probieren Sie einmal die unterschiedlichen Filter aus, und sehen Sie sich anschließend die Formeln genauer an.

Auf diese Weise gelangen Sie leicht zu komplexeren Filtern wie diesem hier: alle Flüge im dritten oder vierten Quartal des Jahres 2019 mit Abflug nach 18:00 Uhr.

```
(
  (DatePart("q", [AbflugDatum]) = 3)
  Or
  (DatePart("q", [AbflugDatum]) = 4)
)
And
(Year([AbflugDatum]) = 2019)
And
([AbflugZeit] > #18:00#)
```

In den Materialien zum Buch finden Sie dieses Beispiel in der Tabelle *tblFlug* in der Datenbank *02_Access_als_Datenbank_Tabellen\2.5.9_Fluege_Filter.accdb*.

Bitte fragen Sie mich nicht, warum Access im Filter ausschließlich englische Operatoren (And, Or) und Funktionsnamen (Year, DatePart) akzeptiert. An anderen Stellen (u. a. in Gültigkeitsregeln) können Sie sowohl die deutsche als auch die englische Bezeichnung verwenden, wobei die deutsche Version von Access die Formel beim Abspeichern ins Deutsche übersetzt. Im Falle des Filters ist das jedoch nicht so, und Sie dürfen nur die englische Schreibweise verwenden.

Zusammenfassung

In diesem Abschnitt habe ich Ihnen das Konzept des Filterns vorgestellt. Es ist die übliche Vorgehensweise, Daten in Access zu suchen und weiterzuverarbeiten. Wir haben uns bisher nur mit dem Filtern in der Datenblattansicht einer Tabelle beschäftigt. Access bietet hier eine ganze Reihe unterschiedlicher Filter an. Diese Filter können Ihnen die tägliche Arbeit mit Access einfacher machen. Technisch gesehen laufen alle Filter auf das Gleiche hinaus: Eine gültige Formel, die in der Tabelleneigenschaft FILTER eingetragen wird. Beim Speichern der Tabelle werden die Filtereinstellungen (die Formel) abgespeichert. Sie können einen Filter einschalten und wieder ausschalten. Wichtig ist nur, dass Sie sich immer darüber im Klaren sind, ob gerade alle Datensätze einer Tabelle angezeigt werden oder ob ein Filter aktiv ist.

Viele Anwender empfinden Filter auf Tabellenebene als störend

Wer einmal einen aktiven Filter übersehen hat und verzweifelt nach den gefilterten Datensätzen gesucht hat, wird das Filtern auf Tabellenebene verfluchen. Puristen in Sachen Datenbankdesign verzichten komplett auf das Sortieren und Filtern in Tabellen. Schließlich eignen sich für diese Aufgaben hervorragend die Auswahlabfragen.

Obwohl ich mich auch eher zu den Puristen zähle, schätze ich die Filter- und Sortierfunktion auf Tabellenebene doch sehr. Meiner Erfahrung nach ist die Umsetzung in Access sogar führend verglichen mit anderen Datenbankwerkzeugen. Mit anderen Worten: In Access können Sie richtig schnell verschiedene Filter erstellen und anpassen.

Wenn möglich, vermeide ich es aber, einen Filter oder eine Sortierung für eine Tabelle abzuspeichern. Ich empfehle Ihnen dies ebenfalls, damit Sie die Gefahr vermeiden, einen aktiven Filter zu übersehen. Für alle Filter oder Sortierungen, die Sie dauerhaft speichern möchten, erstellen Sie besser eine Auswahlabfrage. Aber für das temporäre Filtern und Sortieren ist die Datenblattansicht einer Tabelle wirklich bestens geeignet.

Kapitel 3

Daten filtern, sortieren und zusammenfassen: Abfragen

Abfragen sind das Werkzeug, mit dem Sie Daten aus mehreren Tabellen zusammenbringen und auswerten.

Abfragen zeigen die Daten aus einer oder mehreren Tabellen an. Ich werde mit einer *einfachen Abfrage* beginnen, die Daten aus nur einer einzigen Tabelle anzeigt. Selbstverständlich können Sie die Tabelle auch direkt in der Datenblattansicht öffnen und alle Daten darin sehen. Eine Abfrage können Sie daher als eine zusätzliche Sicht auf die Tabelle verstehen – deshalb werden Abfragen in anderen Systemen (z. B. Server-Datenbanken) auch *Sichten* (englisch *views*) genannt. Eine Abfrage kann sich in einigen Punkten von der Datenblattansicht einer Tabelle unterscheiden:

1. Anordnung der Felder
2. Benennung der Felder
3. Filtern
4. Sortieren der Datensätze

In Access sind Abfragen *das* Werkzeug zum Filtern und Sortieren. In den nächsten Abschnitten werde ich Ihnen alle diese vier Punkte im Detail erklären.

Abgrenzung zwischen Tabellen und Abfragen

Ich empfehle Ihnen, diese logische Trennung in der Arbeitsweise mit Tabellen und Abfragen zu beherzigen:

- ▶ **Tabellen:** Erfassen, Ändern und Löschen von Daten
- ▶ **Abfragen:** Auswerten von Daten (Filtern, Sortieren, Berechnen, Zusammenfassen)

Dadurch wird Ihnen die Arbeit des Datenbankdesigns erheblich einfacher fallen!

3.1 Einfache Abfragen

Bevor wir loslegen, noch ein paar Worte zum Filtern. Es gibt zwei Konzepte, die zum Filtern gehören:

1. Vertikal filtern

Hiermit ist gemeint, dass von den verfügbaren Feldern nur bestimmte *Felder* angezeigt werden sollen. In der Datenblattansicht werden Felder als Spalten dargestellt. Daher entspricht diese Art des Filterns einer Auswahl bestimmter Spalten. Entsprechend auch der Begriff, *vertikal* zu filtern.

2. Horizontal filtern

Datensätze werden in der Datenblattansicht als Zeilen dargestellt. Folglich spricht man vom *horizontalen* Filtern, wenn nur bestimmte *Datensätze* angezeigt werden sollen.

Gleichwohl können Sie in einer Abfrage gleichzeitig vertikal und horizontal filtern. Ich zeige Ihnen aber beides zunächst getrennt voneinander.

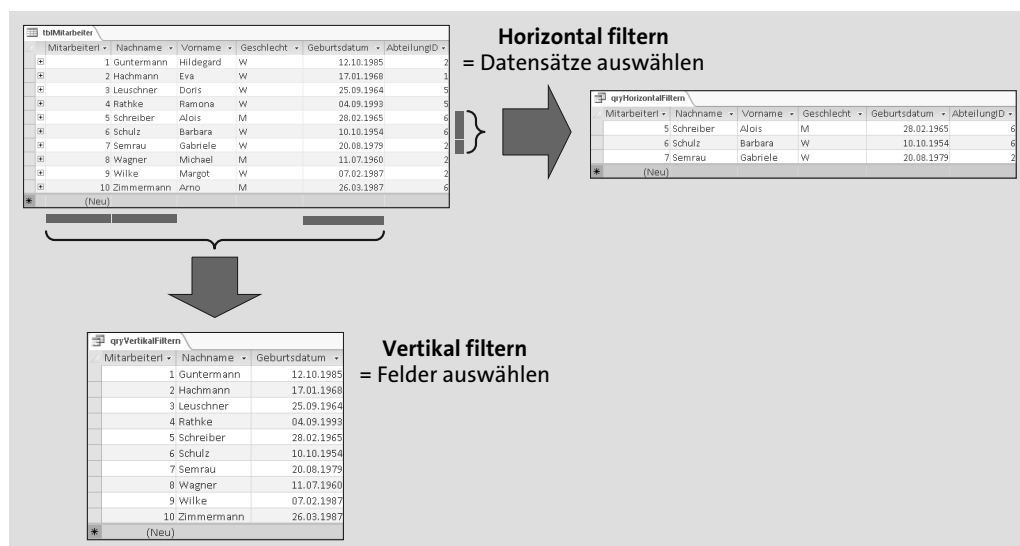


Abbildung 3.1 Es gibt zwei Konzepte zum Filtern: das vertikale Filtern zur Auswahl bestimmter Felder (Spalten) und das horizontale Filtern zur Auswahl bestimmter Datensätze.

Filter und Sortierung in Tabellen vs. Abfragen

Wie Sie in einer Tabelle filtern (und sortieren), habe ich Ihnen bereits gezeigt. Warum also das Rad mit Abfragen neu erfinden? Bei Abfragen fallen mir zwei entscheidende Vorteile ein:

- Sie schaffen eine logische Trennung innerhalb der Datenbank (Tabelle = Datenspeicher; Abfragen = Filtern, Sortieren, Auswerten).
- Für eine Tabelle können Sie mehrere Abfragen erstellen (= mehrere Filter, Sortierungen etc. speichern).

Trotzdem ist Filtern und Sortieren in einer Tabelle erlaubt.

3.1.1 Vertikales Filtern – Felder auswählen

Wir erstellen nun einen Filter für die Tabelle *tblMitarbeiter* unserer Beispieldatenbank *O3_Abfragen\3_Fluege.accdb*. Das Ziel soll eine Abfrage sein, die nur die Felder »Vorname«, »Nachname« und »Geschlecht« anzeigt. Diese Ansicht wäre als öffentliche Liste aller Mitarbeiter gut geeignet, denn viele Personen empfinden das Geburtsdatum als Privatangelegenheit, die in einem Unternehmen höchstens die Personalabteilung etwas angeht.

So erstellen Sie die neue Abfrage zum vertikalen Filtern:

1. Öffnen Sie die Datenbank *O3_Abfragen\3_Fluege.accdb*.
 2. Klicken Sie auf ERSTELLEN • ABFRAGEENTWURF.
- Eine leere Abfrage mit dem Dialog TABELLE ANZEIGEN sollte automatisch erscheinen. Falls Sie das Fenster versehentlich geschlossen haben, klicken Sie auf ENTWURF • TABELLE ANZEIGEN.
3. Wählen Sie *tblMitarbeiter* aus, und klicken Sie auf HINZUFÜGEN.
 4. Schließen Sie den Dialog TABELLE ANZEIGEN (indem Sie auf SCHLIESSEN klicken).

Im oberen Bereich der Abfrage-Entwurfsansicht ist jetzt die Tabelle *tblMitarbeiter* mit ihren Feldern zu sehen (Abbildung 3.2).

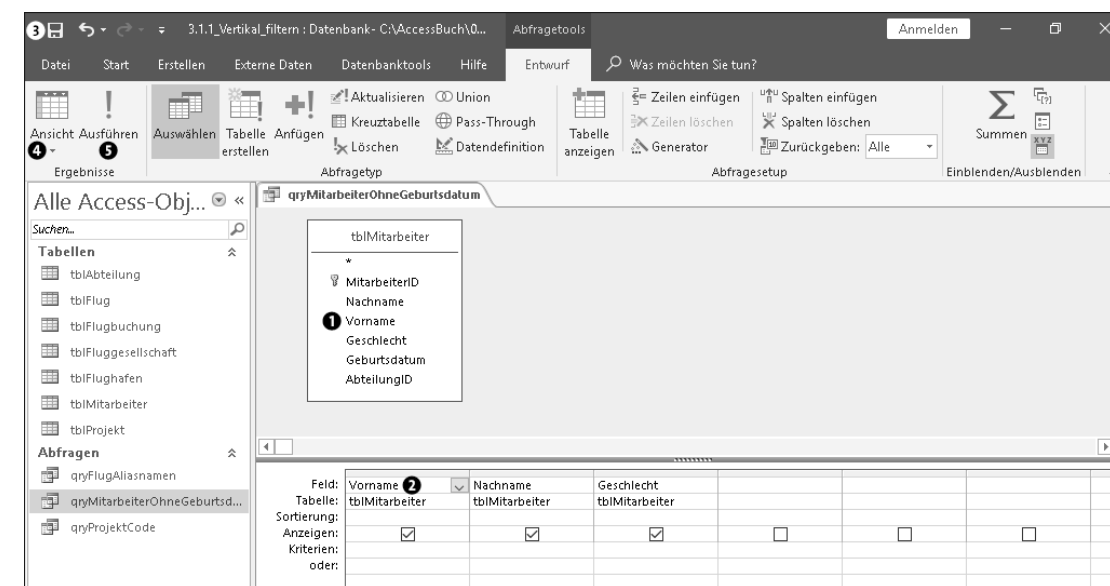


Abbildung 3.2 Die Entwurfsansicht einer Abfrage – das ist der grafische Abfrage-Editor. Im oberen Bereich ist die Tabelle zu sehen, aus der die Daten kommen. Im unteren Bereich stehen die Felder, die Bestandteil der Abfrage sind.

5. Ziehen Sie das Feld »Vorname« ① per Drag & Drop in den unteren Bereich ② (alternativ: Doppelklick auf das Feld).

- Ziehen Sie ebenso die Felder »Nachname« und »Geschlecht« in den unteren Bereich ❷.
- Klicken Sie in der Symbolleiste für den Schnellzugriff auf **SPEICHERN** ❸ (oder **Strg** + **S**), und geben Sie als Name der Abfrage *qryMitarbeiterOhneGeburtsdatum* an.
- Schalten Sie in die Datenblattansicht um, indem Sie unter **ENTWURF • ERGEBNISSE • ANSICHT** ❹ den Eintrag **DATENBLATTANSICHT** auswählen (oder **ENTWURF • ERGEBNISSE • AUSFÜHREN** ❺).

In der Datenblattansicht der Abfrage sehen Sie das Ergebnis des Filterns. Nur die drei gewählten Felder werden angezeigt. In den Materialien zum Buch finden Sie die Abfrage *qryMitarbeiterOhneGeburtsdatum* in der Datenbank *O3_Abfragen\3.1.1_Vertikal_filtern.accdb*.

Ansicht einer Abfrage	Abfragedefinition verändern	Datensätze filtern und sortieren	Daten ändern
Entwurfsansicht	•	•	
Datenblattansicht		•	•
SQL-Ansicht	•	•	

Tabelle 3.1 Access kennt drei Ansichten für Abfragen. Die Datenblattansicht kennen Sie bereits von den Tabellen. In diesem Kapitel werden wir mit der Entwurfsansicht arbeiten.

Es zählt die Reihenfolge der Felder in der Entwurfsansicht der Abfrage

In der Entwurfsansicht der Abfrage sehen Sie die ausgewählten Felder nebeneinander angeordnet. Die Reihenfolge ist nicht egal! Die Spalten in der Datenblattansicht der Abfrage werden genau so angezeigt, wie sie in der Entwurfsansicht angeordnet sind. Dies eröffnet Ihnen eine hervorragende Möglichkeit: Sie können die Felder in einer Abfrage nach Belieben anordnen und verschieben! Die Reihenfolge der Felder in der Tabelle wird nicht verändert und spielt auch keine Rolle.

So können Sie die Reihenfolge der Felder in einer Abfrage ändern:

- Öffnen Sie die Abfrage in der Entwurfsansicht.

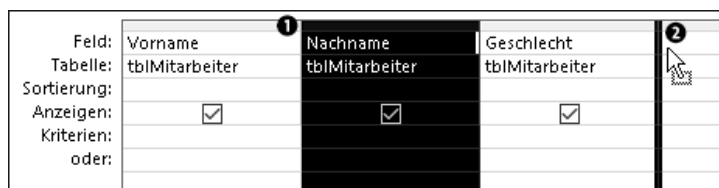


Abbildung 3.3 In der Entwurfsansicht sehen Sie über dem Feldnamen einen schmalen grauen Balken. Anklicken markiert das gesamte Feld.

- Gehen Sie im unteren Bereich zu dem Feld, das Sie verschieben möchten.

- Klicken Sie auf den schmalen grauen Balken ❶ über dem Feldnamen. Das Feld ist jetzt markiert (schwarzer Hintergrund).
- Ziehen Sie das Feld per Drag & Drop auf dem grauen Balken an die neue Position ❷.

Felder ausblenden oder entfernen?

Ihnen sind sicherlich schon die Häkchen in der Zeile **ANZEIGEN** aufgefallen. Wenn Sie das Häkchen für ein Feld wegnehmen, erscheint dieses Feld auch nicht mehr in der Datenblattansicht. Das Feld ist aber immer noch Bestandteil der Abfrage!

Wozu benötigt man Felder, die nicht angezeigt werden? In den nächsten beiden Abschnitten werde ich Ihnen zeigen, wie Sie Datensätze filtern und sortieren. Wir werden sehen, dass Sie dazu auch Felder verwenden können, die nicht in der Ausgabe sichtbar sein sollen. Genau dafür gibt es die Möglichkeit, das Häkchen bei **ANZEIGEN** zu entfernen.

Wenn Sie ein Feld in einer Abfrage gar nicht verwenden möchten, ist es sinnvoller, dieses Feld aus der Abfrage zu entfernen.

- Öffnen Sie die Abfrage in der Entwurfsansicht.
- Markieren Sie das gewünschte Feld per Mausklick auf den schmalen grauen Balken.
- Klicken Sie auf **ENTWURF • ABFRAGESETUP • SPALTEN LÖSCHEN** (oder drücken Sie **Entf**).

Sie sehen sowohl in der Entwurfs- als auch in der Datenblattansicht sofort, dass das entfernte Feld nicht mehr Bestandteil der Abfrage ist. Die Tabelle (dargestellt im oberen Bereich der Entwurfsansicht) ist davon hingegen nicht betroffen. Dort gibt es das Feld nach wie vor, und Sie können es, wenn notwendig, per Drag & Drop wieder zur Abfrage hinzufügen.

Sie können Felder auch mehrfach anzeigen lassen

Probieren Sie einmal aus, in der Entwurfsansicht einer Abfrage ein Feld zweimal per Drag & Drop in den unteren Bereich zu ziehen. Das funktioniert tadellos! Sie können in einer Abfrage ein Feld doppelt oder mehrfach anzeigen lassen.

In der Datenblattansicht bekommt eines der Felder automatisch einen neuen Namen: »Expr1000«. Dies ist ein automatisch vergebenen *Alias*. Besser ist es jedoch, einen aussagekräftigen *Alias* selbst festzulegen.

Aliasse für Felder

Was ist ein *Alias*, und in welchen Fällen benötigt man einen? Die englische Bezeichnung *alias* bedeutet so etwas wie Pseudonym. In einer Abfrage können Sie dadurch einen Feldnamen nur für die Anzeige ändern, also ohne dass sich etwas an der Tabelle ändert.

In der Tabelle *tblProjekt* gibt es das Feld »ProjektAbkürzung«. Nehmen wir an, Sie möchten in einer Abfrage den Feldnamen in »ProjektCode« ändern. Das geht in der Entwurfsansicht einer Abfrage ganz einfach:

1. Erstellen Sie eine neue Abfrage in der Entwurfsansicht.
2. Fügen Sie die Tabelle *tblProjekt* hinzu.
3. Ziehen Sie das Feld »ProjektID« per Drag & Drop in den unteren Bereich (alternativ: Doppelklick auf das Feld).
4. Ziehen Sie das Feld »ProjektAbkuerzung« ebenfalls in den unteren Bereich.
5. Klicken Sie in der Zeile FELD vor den Feldnamen »ProjektAbkuerzung«.
6. Tragen Sie den Alias gefolgt von einem Doppelpunkt ein:
ProjektCode:
7. Speichern Sie die Abfrage unter dem Namen *qryProjektCode*.

Diese Abfrage finden Sie in den Materialien zum Buch ebenfalls in der Datenbank *O3_Abfragen\3.1.1_Vertikal_filtern.accdb*. Wenn Sie in die Datenblattansicht wechseln, werden Sie sehen, dass statt des Feldnamens der Alias angezeigt wird. Der Doppelpunkt hinter dem Alias, den Sie in der Entwurfsansicht eingetragen haben, erscheint jedoch nicht.

Wozu benötigt man einen Alias?

Es gibt einige typische Fälle, in denen Aliasse üblicherweise verwendet werden:

- ▶ Ein Feld soll mehrfach angezeigt werden.
- ▶ Die Bedeutung eines Feldes hat sich z. B. durch einen Filter geändert.
- ▶ Es handelt sich um ein berechnetes Feld (mehr dazu in Abschnitt 3.2, »Auswerten von Daten eines Datensatzes: Berechnete Felder«).
- ▶ Es handelt sich um ein zusammengefasstes Feld (mehr dazu in Abschnitt 3.3, »Zusammenfassen von Datensätzen: Gruppierung und Aggregieren«).
- ▶ Bei Abfragen mit mehreren Tabellen (siehe Abschnitt 3.4, »Abfragen von Daten aus mehreren Tabellen«) sind die Feldnamen häufig nicht mehr eindeutig oder genügend aussagekräftig.

Beim Programmieren mit Access werden Ihnen sicherlich noch weitere Einsatzgebiete für Aliasse auffallen. Wichtig ist, dass Sie stets wissen, woher die Daten kommen – wie also das ursprüngliche Feld in der Tabelle heißt. Ein Blick in die Entwurfsansicht der Abfrage hilft Ihnen dabei weiter.

Ein weiteres Beispiel mit mehreren Aliassen sehen Sie in der Abfrage *qryFlugAliasse*, die Sie in den Materialien zum Buch in der Datenbank *O3_Abfragen\3.1.1_Vertikal_filtern.accdb* finden.

Lassen Sie uns noch einmal zurückgehen zu einem Feld, das mehrfach in einer Abfrage vorkommt. Mit einem Alias können Sie sinnvolle Bezeichnungen für beide Spalten festlegen. In der Datenblattansicht werden Sie feststellen, dass die Inhalte in beiden Spalten gleich sind. Interessant wird es, wenn Sie einen Wert in einer Spalte ändern (Abbildung 3.4):

Vorname	AliasVorname	Nachname	Geschlecht
Hildegard	Hildegard	Guntermann	W
Eva	Eva GEÄNDERT	Hachmann	W
Doris	Doris	Leuschner	W
Ramona	Ramona	Rathke	W

Vorname	AliasVorname	Nachname	Geschlecht
Hildegard	Hildegard	Guntermann	W
Eva GEÄNDEF	Eva GEÄNDERT	Hachmann	W
Doris	Doris	Leuschner	W
Ramona	Ramona	Rathke	W

Abbildung 3.4 In einer Abfrage können Sie ein Feld mehrfach anzeigen lassen. Nach dem Abspeichern des Datensatzes sehen Sie die Änderungen auch in dem anderen Feld.

Die Änderung sehen Sie im anderen Feld erst nach dem Abspeichern des Datensatzes (oder wenn Sie in das andere Feld klicken). Letztendlich basieren die beiden Spalten auf demselben Feld. Daher können sich die Werte nicht unterscheiden!

Abfragen können keine Inhalte speichern

Vergessen Sie bitte nicht, dass Abfragen nur eine andere Sicht auf eine Tabelle sind. Mit vielen Abfragen können Sie Daten ändern und Datensätze hinzufügen. Gespeichert werden die Daten aber immer noch in der Tabelle (oder den Tabellen), die der Abfrage zugrunde liegen. Das hat wichtige Konsequenzen. Wenn Sie zusätzliche Felder erfassen möchten, *müssen Sie die Tabellenstruktur ändern!*

In der Entwurfsansicht einer Abfrage kommen Sie nicht weiter. Hier können Sie zwar ein Feld mehrfach anzeigen lassen. Die Inhalte sind aber identisch, denn letztendlich ist es immer noch das eine Feld aus der Tabelle.


Das Sternchen in Abfragen

In der Entwurfsansicht einer Abfrage sehen Sie in der Tabelle, die im oberen Bereich eingebildet wird, in der ersten Zeile ein Sternchen. Das Sternchen hat in Abfragen eine besondere Bedeutung: Es steht für »alle Felder«.

Wenn Sie alle Felder einer Tabelle in der Abfrage anzeigen lassen möchten, haben Sie dazu zwei Möglichkeiten:

1. **Sie ziehen jedes Feld per Drag & Drop nach unten.**

Der Vorteil ist, dass Sie anschließend die Reihenfolge der Felder ändern und Aliasse vergeben können. Außerdem könnten Sie gleich mit dem horizontalen Filtern und dem Sortieren loslegen.

Übrigens müssen Sie nicht jedes Feld einzeln hintereinander nach unten ziehen. Klicken Sie auf das erste Feld, halten Sie die Taste  gedrückt, und klicken Sie auf das letzte Feld der Tabelle. Dadurch haben Sie alle Felder markiert und können sie gleichzeitig nach unten ziehen.

2. Sie ziehen nur das Sternchen per Drag & Drop nach unten.

Das geht zum einen sehr schnell. Die Reihenfolge der Felder wird dabei aus der Tabelle übernommen. Zum anderen werden alle Änderungen der Tabellenstruktur automatisch berücksichtigt. Wenn Sie beispielsweise ein Feld zur Tabelle hinzufügen, erscheint dieses neue Feld automatisch in allen Abfragen, die das Sternchen verwenden.

In der Praxis nutze ich beide der Varianten gleichermaßen. In den Beispielen in den Materialien zum Buch können Sie sich ein Bild davon machen, in welchen Fällen die eine oder die andere Möglichkeit besser geeignet ist.

3.1.2 Horizontales Filtern – Datensätze auswählen

Beim horizontalen Filtern werden nur bestimmte Datensätze angezeigt. Der einfachste Filter dieser Art ist die Anzahl der Datensätze.

Die maximale Anzahl der Datensätze festlegen

In der Entwurfsansicht einer Abfrage können Sie unter ENTWURF • ABFRAGESETUP • ZURÜCKGEBEN einstellen, wie viele Datensätze maximal angezeigt werden dürfen.

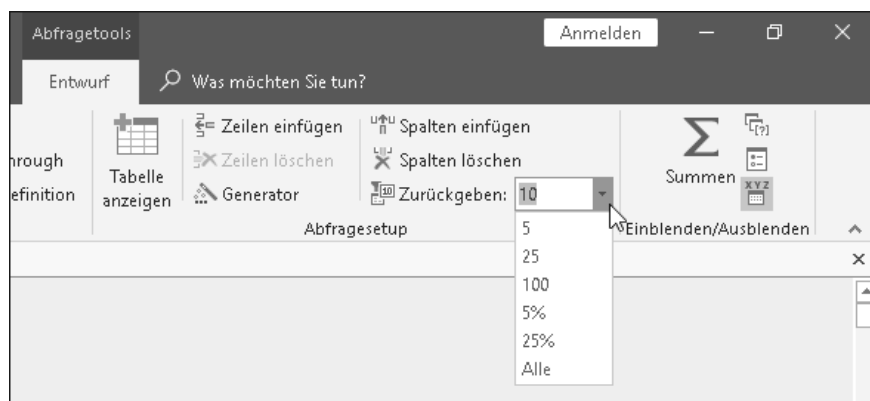


Abbildung 3.5 Über die Einstellung »Zurückgeben« können Sie festlegen, wie viele Datensätze maximal angezeigt werden dürfen.

Standardmäßig werden alle Datensätze zurückgegeben (Einstellung ALLE). Wenn Sie in das Feld ZURÜCKGEBEN z. B. die Zahl 10 eintragen, werden nur die ersten zehn Datensätze angezeigt. Ebenso können Sie eine der vorgegebenen Zahlen aus der Dropdown-Liste auswählen.

Wie Sie in der Dropdown-Liste sehen, gibt es als zweite Form der Angabe die Prozentzahlen. Auf diese Weise werden beispielsweise die ersten 30 % aller Datensätze zurückgegeben. Wenn die Tabelle insgesamt 50 Datensätzen enthält, werden in diesem Fall nur 15 Datensätze angezeigt.

Beispiele zu dieser recht einfachen Form des horizontalen Filterns finden Sie in den Materialien zum Buch unter *O3_Abfragen\3.1.2_TOP.accdb*.

Ein Tipp zum Feld »Zurückgeben«

Wenn Sie eine unbekannte Abfrage von jemand anderem erhalten, lohnt es sich, einen Blick auf das Feld ZURÜCKGEBEN zu werfen. Wenn hier *nicht* der Eintrag ALLE steht, werden Sie nur einen Teil der gefilterten Datensätze sehen.

Horizontale Filter mit Kriterien

Diese Einstellung – die maximale Anzahl der Datensätze, die angezeigt werden dürfen – ist eigentlich nur eine kleine Zugabe. Denn das Wesentliche zum vertikalen Filtern sind *Kriterien*, die Sie in der Entwurfsansicht einer Abfrage im unteren Bereich festlegen können. Sie können Kriterien für ein oder mehrere Felder vorgeben. Für jeden Datensatz entscheidet Access anhand der von Ihnen festgelegten Kriterien, ob er angezeigt oder ausgefiltert werden soll.

Die Filterkriterien sehen je nach Felddatentyp unterschiedlich aus. Ich werde Ihnen für die wichtigsten Felddatentypen die Filterkriterien anhand von Beispielen erläutern. Alle Abfragen finden Sie in der Datenbank *O3_Abfragen\3.1.2_Horizontal_filtern.accdb* in den Materialien zum Buch.

Filterkriterien für Textfelder

Beginnen wir zunächst mit einem ganz einfachen Textfilter. In der Tabelle *tblMitarbeiter* enthält das Feld »Geschlecht« entweder »M« oder »W«. Um eine Liste aller männlichen Mitarbeiter zu erhalten, filtern Sie alle Datensätze mit dem Wert »M« im Feld »Geschlecht«. Und das geht so:

1. Öffnen Sie die Datenbank *O3_Abfragen\3_Fluege.accdb*.
2. Erstellen Sie eine neue Abfrage in der Entwurfsansicht.
3. Fügen Sie die Tabelle *tblMitarbeiter* hinzu.
4. Ziehen Sie alle Felder der Tabelle *tblMitarbeiter* per Drag & Drop in den unteren Bereich (alternativ: Doppelklick auf die Felder).
5. Tragen Sie für das Feld »Geschlecht« in der Zeile KRITERIEN "M" ein.

6. Speichern Sie die Abfrage unter dem Namen *qryMitarbeiterMaennlich*, und schalten Sie in die Datenblattansicht um.

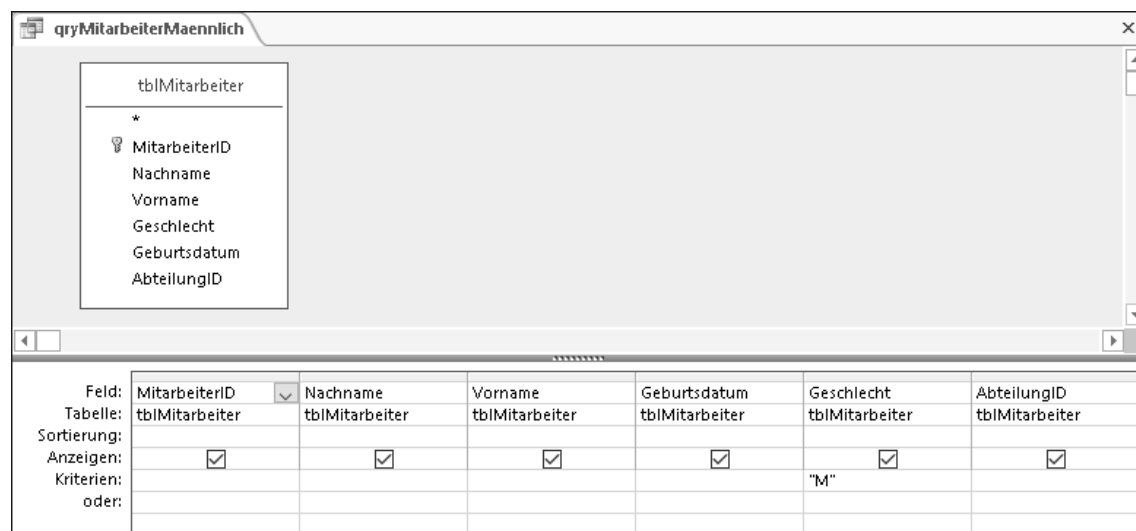


Abbildung 3.6 Ein sehr einfacher Textfilter: alle Mitarbeiter mit dem Wert »M« im Feld »Geschlecht« (= alle Männer)

Ebenso erhalten Sie eine Liste aller Mitarbeiterinnen, indem Sie alle Datensätze mit dem Wert »W« im Feld »Geschlecht« filtern (Abfrage *qryMitarbeiterWeiblich*). Bei einem einfachen Textfilter zeigt Access nur diejenigen Datensätze an, bei denen der Feldinhalt vollständig mit dem Kriterium übereinstimmt, genauer gesagt mit dem Text innerhalb der Anführungszeichen.

»Ist gleich« und »ist ungleich«

Wenn Sie möchten, können Sie in diesem Beispiel das Gleichheitszeichen im Kriterium explizit angeben:

= "M"

Ebenso gibt es das Ungleichheitszeichen:

<> "M"

Alternativ können Sie statt des Ungleichheitszeichens den Operator *Nicht* einsetzen:

Nicht "M"

Als zweites Beispiel zeige ich Ihnen einen Filter mit einer Liste von Werten. Beispielsweise sollen alle Mitarbeiter angezeigt werden, die »Ramona«, »Margot«, »Michael« oder »Alois« heißen.

- Öffnen Sie die Datenbank *O3_Abfragen\3_Fluege.accdb*.
- Erstellen Sie eine neue Abfrage in der Entwurfsansicht.
- Fügen Sie die Tabelle *tblMitarbeiter* hinzu.
- Ziehen Sie das Feld mit dem Sternchen per Drag & Drop in den unteren Bereich (alternativ: Doppelklick auf das Sternchen).
- Ziehen Sie das Feld »Vorname« per Drag & Drop in den unteren Bereich.
- Entfernen Sie für das Feld »Vorname« das Häkchen bei ANZEIGEN.

An dieser Stelle eine kleine Anmerkung: Diesmal sind über das Sternchen bereits alle Spalten der Tabelle *tblMitarbeiter* in der Abfrage enthalten. Für den Filter ist das Feld »Vorname« separat notwendig (siehe nächster Schritt). Damit das Feld in der Datenblattansicht nicht doppelt angezeigt wird, entfernen wir das Häkchen bei ANZEIGEN.

- Tragen Sie für das Feld »Vorname« in der Zeile KRITERIEN

In ("Ramona";"Margot";"Michael";"Alois")

ein. Für längere Kriterien kann es angenehmer sein, das ZOOM-Fenster zu verwenden. Gehen Sie hierzu in das Kriterienfeld, und drücken Sie \uparrow + $F2$.

- Speichern Sie die Abfrage unter dem Namen *qryMitarbeiterNamenAusListe*, und schalten Sie in die Datenblattansicht um.

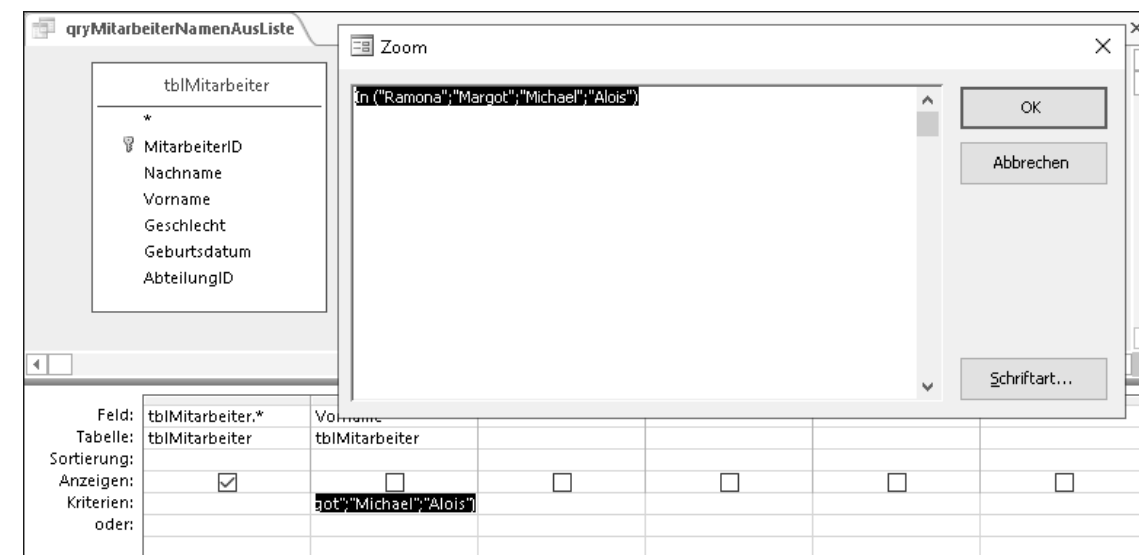


Abbildung 3.7 Längere Kriterien lassen sich bequemer über das »Zoom«-Fenster eintragen.

Auch beim Filtern mit einer Liste werden nur diejenigen Datensätze angezeigt, bei denen der Feldinhalt vollständig mit einem Wert aus der Liste übereinstimmt.

Es gibt aber auch Textfilter, bei denen Sie nur einen Teil des Feldinhalts festlegen, beispielsweise ein Filter aller Mitarbeiter, deren Nachname mit dem Buchstaben S beginnt. Dazu können Sie *Textmustervergleiche* mit den Operatoren *Wie* und *ALike* verwenden, die Sie in Abschnitt 2.3.12, »Feldeinschränkungen«, kennengelernt haben. Tabellen mit den verschiedenen Platzhaltern für beide Operatoren finden Sie dort. Der wichtigste Platzhalter für den Operator *Wie* ist wahrscheinlich das Sternchen, das für eine beliebige Zeichenfolge steht (bitte nicht verwechseln mit dem Sternchen in Abfragen, das für »alle Felder« steht). Um nur Mitarbeiter anzuzeigen, deren Nachname mit dem Buchstaben S beginnt, verwenden Sie für das Feld »Nachname« das Kriterium

Wie "S*"

In der Beispieldatenbank habe ich diese Abfrage unter *qryMitarbeiterS* gespeichert. Hier noch ein paar weitere Beispiele für Filter mit Textmustervergleichen:

- ▶ Der Nachname beginnt nicht mit dem Buchstaben S:
Nicht Wie "S*"
- ▶ Der Nachname beginnt mit den Buchstaben R, S oder T:
Wie "[RST]*"
- ▶ Der Nachname endet auf »mann«:
Wie "*mann"
- ▶ Der Nachname enthält den Buchstaben h:
Wie "*h*"

Die Access-Einstellung »SQL Server-kompatible Syntax«

Normalerweise unterscheidet Access zwischen den Operatoren *Wie* (englisch *Like*) und *ALike* (*ANSI Like*). Von der Syntax her ist *ALike* sehr nahe dem *Like*-Operator des Microsoft SQL Servers. Unter anderem verwendet *ALike* das Prozentzeichen % als Platzhalter für eine beliebige Zeichenfolge (mehr dazu in Abschnitt 2.3.12, »Feldeinschränkungen«). Ob Sie eher den Operator *Wie* oder lieber *ALike* verwenden, ist Geschmackssache.

Es gibt aber eine entscheidende Einstellung in den Access-Optionen, die unter anderem den Operator *Wie* betrifft. In den Access-Optionen unter **OBJEKT-DESIGNER • ABFRAGEENTWURF • SQL SERVER-KOMPATIBLE SYNTAX (ANSI 92)** ist die Option **IN DIESER DATENBANK VERWENDEN** standardmäßig deaktiviert. Wenn Sie hier ein Häkchen setzen, bekommt der Operator *Wie* die gleiche Bedeutung wie der Operator *ALike*! Das erkennen Sie auch daran, dass das Wort »Wie« von Access automatisch durch »ALike« ersetzt wird.

Wenn Sie die *SQL-Server-kompatible Syntax* für eine Datenbank aktivieren, hat das weitreichende Folgen, auf die Access Sie auch hinweist (Abbildung 3.8). Unter anderem müssen Sie alle Textmuster anpassen (Prozentzeichen anstelle des Sternchens). Neben den Textmustern hat diese Einstellung aber noch andere Auswirkungen. Darauf werde ich Sie an entsprechen-

der Stelle hinweisen. Bevor Sie die SQL-Server-kompatible Syntax aktivieren, sollten Sie genau wissen, welche Bereiche Ihrer Datenbank davon betroffen sind. Je nach Größe der Datenbank ist diese Umstellung keine Sache, die sich in ein paar Minuten nebenbei erledigen lässt!

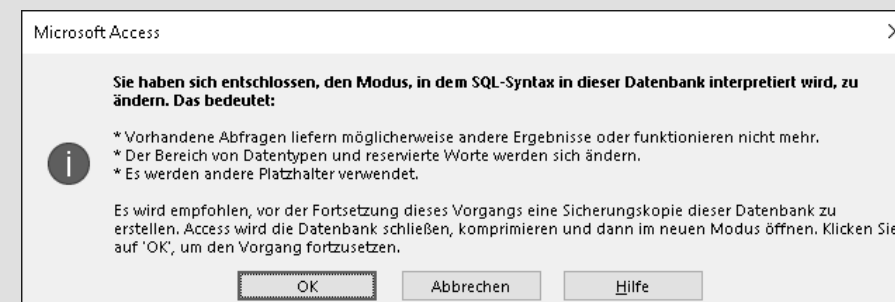


Abbildung 3.8 Die SQL-Server-kompatible Syntax hat weitreichende Auswirkungen in Ihrer Datenbank – nicht nur auf die Abfragen.

Filterkriterien für den Felddatentyp »Ja/Nein«

Ein Filter für ein JA/NEIN-Feld kann entweder auf wahr (Häkchen gesetzt) oder falsch (Häkchen nicht gesetzt) prüfen. Tragen Sie im Kriterium einfach die Zeichenfolge *Wahr* bzw. *Falsch* ein – und zwar *ohne* Anführungszeichen.

Es gibt Flughäfen, die nicht mehr aktiv sind. Beispielsweise ist das Flugfeld von Berlin-Tempelhof mittlerweile ein Park. Hier also bitte nicht mehr landen! Im Feld »IstAktiv« der Tabelle *tblFlughafen* ist das Häkchen daher nicht gesetzt. Eine Liste der aktiven Flughäfen erhalten Sie über die Abfrage *qryFlughafenAktiv*:

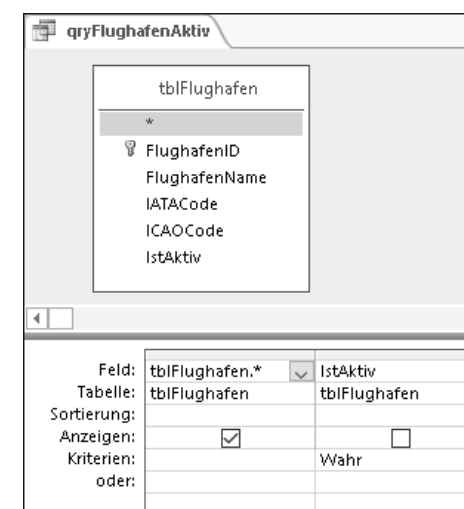


Abbildung 3.9 Ein Filter der Flughäfen, die aktiv sind

Filterkriterien für Zahlenfelder

Alle Zahlenfelder – egal, ob Felddatentyp ZAHL oder WÄHRUNG – können Sie für Filterkriterien heranziehen. Hier zwei Beispiele für das Feld »Kosten« in der Tabelle *tblFlugbuchung*:

- ▶ Flugbuchungen teuer als 500 €:
>500
- ▶ Flugbuchungen zwischen 300 € und 500 €, beide Werte inklusive:
Zwischen 300 Und 500

Filterkriterien für den Felddatentyp »Datum/Uhrzeit«

Felder mit dem Felddatentyp DATUM/UHRZEIT werden von Access intern ebenfalls als Zahlen gespeichert. Entsprechend können Sie auch solche Felder filtern. Als Beispiel filtern wir Mitarbeiter nach dem Geburtsdatum.

Zunächst erstellen wir eine Abfrage, die alle Mitarbeiter ausgibt, die vor 1980 geboren sind. Wie in Abschnitt 2.3.12, »Feldeinschränkungen«, erläutert, können Sie feste Datumsangaben eingerahmt von zwei Hash-Zeichen (#) angeben. Tragen Sie daher für das Feld »Geburtsdatum« als Filterkriterium

```
<#01.01.1980#
```

ein. In der Datenbank *O3_Abfragen\3.1.2_Horizontal_filtern.accdb* in den Materialien zum Buch finden Sie diese Abfrage unter dem Namen *qryMitarbeiterGeborenVor1980*.

Mit Funktionen können Sie Filterkriterien für den Felddatentyp »Datum/Uhrzeit« erweitern

In Abschnitt 3.2, »Auswerten von Daten eines Datensatzes: Berechnete Felder«, werde ich Ihnen eine ganze Reihe von *Funktionen* vorstellen. Sie können Funktionen auch in Filterkriterien einsetzen. Gerade bei Feldern mit dem Felddatentyp DATUM/UHRZEIT ist das eine echte Bereicherung. Daher möchte ich an dieser Stelle ein Beispiel vorwegnehmen:

Mitarbeiter, die derzeit nicht älter als 30 Jahre alt sind:

```
>=DatAdd("jjjj";-30;Jetzt())
```

Dieses Beispiel finden Sie in der Abfrage *qryMitarbeiterJuengerAls30Jahre*. Sowohl auf die Funktion *Jetzt()* als auch auf *DatAdd()* werde ich in Abschnitt 3.2.6, »Funktionen für Datumsfelder«, eingehen.

Andere Felder in Filterkriterien verwenden

Bisher haben wir nur feste Bezugswerte in Filterkriterien verwendet, beispielsweise als Fixdatum den 01.01.1980, der als Kriterium für alle Datensätze herangezogen wurde (Abfrage *qryMitarbeiterGeborenVor1980*). Es gibt aber auch die Möglichkeit, andere Felder im Filterkriterium zu verwenden.

Ich zeige Ihnen das an einem Beispiel mit der Tabelle *tblFlug*. In den Feldern »AbflugZeit« und »AnkunftZeit« sind die Uhrzeiten von Abflug und Landung eingetragen. Selbstverständlich findet die Landung zeitlich gesehen immer nach dem Abflug statt. Trotzdem kann die Uhrzeit im Feld »AnkunftZeit« kleiner als die im Feld »AbflugZeit« sein. Dies sind dann Flüge, die über Nacht gehen und bei denen die Landung erst am nächsten Tag stattfindet. Mit einer Abfrage können wir diese Art von Flügen filtern:

1. Öffnen Sie die Datenbank *O3_Abfragen\3_Fluege.accdb*.
2. Erstellen Sie eine neue Abfrage in der Entwurfsansicht.
3. Fügen Sie die Tabelle *tblFlug* hinzu.
4. Ziehen Sie das Feld mit dem Sternchen per Drag & Drop in den unteren Bereich (alternativ: Doppelklick auf das Sternchen).
5. Ziehen Sie das Feld »AnkunftZeit« per Drag & Drop in den unteren Bereich.
6. Entfernen Sie für das Feld »AnkunftZeit« das Häkchen bei ANZEIGEN.
7. Tragen Sie für das Feld »AnkunftZeit« in der Zeile KRITERIEN
<[AbflugZeit]
ein.
8. Speichern Sie die Abfrage unter dem Namen *qryFlugLandungAmNaechstenTag*, und schalten Sie in die Datenblattansicht um.

Zeichenfolgen und andere Felder in Filterkriterien

Diese Konventionen müssen Sie beachten, wenn Sie Filterkriterien erstellen:

- ▶ Geben Sie Zeichenfolgen in Anführungszeichen an: entweder die einfachen ' ($\boxed{\diamond} + \boxed{\#}$) oder die doppelten " ($\boxed{\diamond} + \boxed{2}$).
- ▶ Setzen Sie die Namen anderer Felder in eckige Klammern: [und], ($\boxed{\text{Alt Gr}} + \boxed{8}$) und ($\boxed{\text{Alt Gr}} + \boxed{9}$).

Optional können Sie bei Feldern den *voll qualifizierten Feldnamen* mit Angabe des Tabellennamens (Name der Datensatzquelle) verwenden:

```
[<Tabellenname>].[<Feldname>]
```

Diese Konventionen gelten neben den Filterkriterien auch für andere Ausdrücke, beispielsweise auf Tabellenebene für Feld- und Tabelleneinschränkungen sowie auf Ebene der Abfragen für berechnete Felder.

Zur Unterstützung beim Erstellen von Filterkriterien können Sie auf den Ausdrucks-Generator zurückgreifen. Gehen Sie hierfür im Feld »AnkunftZeit« in die Zeile KRITERIEN, und klicken Sie im Menüband auf ABFRAGETOOLS • ENTWURF • ABFRAGESETUP • GENERATOR (oder $\boxed{\text{Strg}} + \boxed{\text{F2}}$).

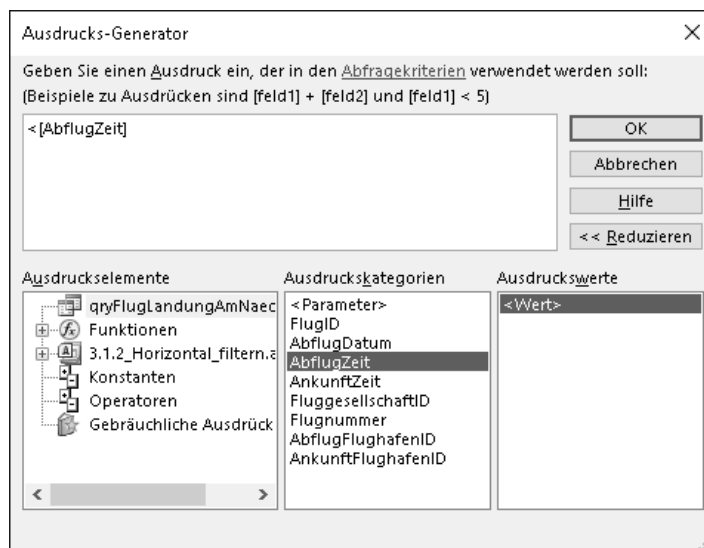


Abbildung 3.10 Mit Hilfe des Ausdrucks-Generators können Sie Filterkriterien bequem per Mausklick zusammenstellen.

Der Ausdrucks-Generator ist recht intuitiv zu bedienen. Hier finden Sie alle verfügbaren Felder, Funktionen, Operatoren und andere nützliche Angaben. Damit können Sie Filterkriterien und andere Ausdrücke bequem per Mausklick zusammenstellen.

Zusammenfassung

Über einen horizontalen Filter legen Sie fest, welche Datensätze in der Datenblattansicht der Abfrage angezeigt werden und welche weggefiltert werden sollen. Zum einen können Sie festlegen, wie viele Datensätze überhaupt zurückgegeben werden dürfen. Interessanter sind die Filterkriterien, anhand derer Sie für ein oder mehrere Felder exakt festlegen können, ob der Datensatz herausgefiltert werden soll oder nicht. Bei den Filterkriterien können Sie auch auf andere Felder zurückgreifen.

Wichtig ist aber, dass jeder Datensatz für sich betrachtet wird. Sie können also nicht auf die Feldinhalte anderer Datensätze zurückgreifen. Mir kommt ein Szenario wie »vergleiche mit dem Feldinhalt aus der Zeile vorher« in den Sinn. So etwas funktioniert mit Abfragen grundsätzlich nicht. Zum einen gibt es den »Datensatz vorher« nicht, denn die Reihenfolge von Datensätzen innerhalb einer Tabelle kann sich jederzeit ändern. Zum anderen wäre dies ein Filter, der Werte aus mehreren Datensätzen verwendet.

Werte aus dem Datensatz vorher oder danach berücksichtigen

In einer richtig normalisierten Datenbank werden Sie selten auf Feldinhalte aus einem vorhergehenden oder folgenden Datensatz zurückgreifen müssen. Falls doch, dann geht das *nicht mit einer Abfrage*.

So ein Szenario lässt sich nur mit fortgeschrittener VBA-Programmierung umsetzen. In Stichpunkten hier die wichtigsten Schritte:

1. in VBA per DAO eine Datensatzquelle öffnen
2. die Sortierung für die Datensatzquelle explizit angeben (ORDER-BY-Klausel)
3. in einer Do-Loop-Schleife durch alle Datensätze iterieren
4. Zwischenwerte gegebenenfalls in lokalen Variablen speichern

Ein Beispiel dazu zeige ich Ihnen im Datenimport-Workshop in Abschnitt 4.5.3, »Workshop: Import einer Tabelle mit fehlenden Daten«.

3.1.3 Umgang mit NULL-Werten

Im letzten Abschnitt habe ich das Thema NULL-Werte ganz bewusst nicht berücksichtigt. Eine recht passende Bezeichnung für einen NULL-Wert ist meines Erachtens »nicht definiert«. Ich werde Ihnen gleich zeigen, warum diese Bezeichnung wichtig ist. Die Frage ist: Wie filtern wir alle Datensätze, die in einem Feld den Wert NULL enthalten?

Dazu müssen wir uns zunächst Felder ansehen, die NULL-Werte enthalten können. In der Tabelle *tblFlugbuchung* sind die Felder »Kosten« und »AbrechnungAufProjektID« optionale Felder und dürfen den Wert NULL enthalten. Wenn Sie sich die Tabelle in der Datenblattansicht ansehen, werden Sie bei den NULL-Werten leere Zellen sehen. Intuitiv denkt man daher an dieses Filterkriterium:

= ""

So funktioniert das leider **nicht**! Dieser Ausdruck prüft auf eine leere Zeichenfolge, was ein definierter Wert ist. NULL bedeutet aber »nicht definiert«, deshalb müssen wir eine andere Strategie anwenden. So sieht es also richtig aus:

NULL-Werte müssen mit speziellen Operatoren geprüft werden

So können Sie überprüfen, ob ein Feld den Wert NULL enthält oder nicht:

Ist Null

Ist Nicht Null

Verwenden Sie *nicht das Gleichheitszeichen*, um auf NULL-Werte zu prüfen!

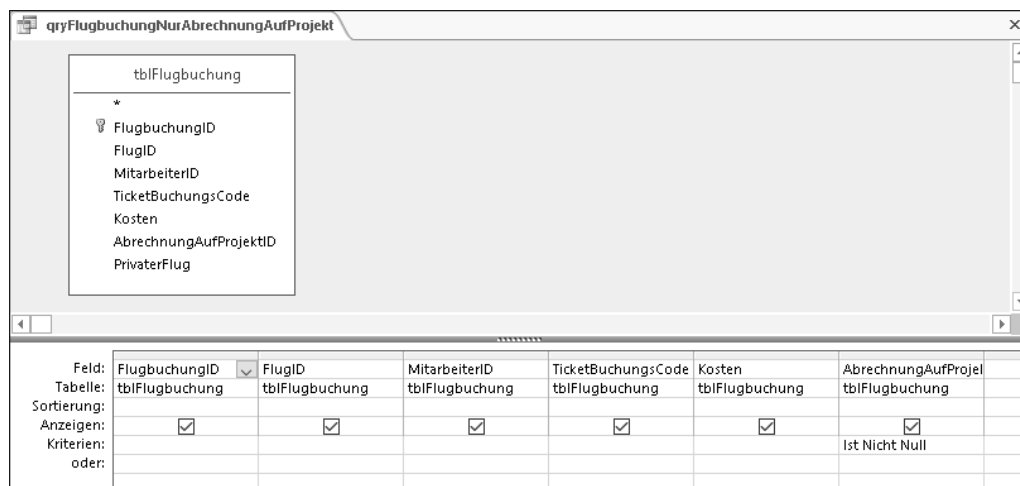


Abbildung 3.11 Dieser Filter gibt nur Flugbuchungen zurück, die auf ein Projekt abgerechnet werden («ProjektID» enthält nicht den Wert NULL).

In den Materialien zum Buch unter *O3_Abfragen\3.1.3_NULL_filtern.accdb* finden Sie diese beiden Beispiele:

1. »**qryFlugbuchungNurAbrechnungAufProjekt**«

Dies ist das beschriebene Beispiel. Einige Reisekosten sollen im Rahmen von Projekten abgerechnet werden. In diesem Fall wird für die Flugbuchung die »ProjektID« vermerkt. Für das Feld »AbrechnungAufProjektID« habe ich als Filterkriterium

Ist Nicht Null
gesetzt.

2. »**qryProjektEnddatumIstNULL**«

Für jedes Projekt in der Tabelle *tblProjekt* wird das Datum vom Beginn und vom Ende des Projekts eingetragen. Wie so häufig sieht man bei manchen Projekten kein Ende ... Für Projekte ohne Enddatum lautet das Filterkriterium für das Feld »Ende« daher

Ist Null

3.1.4 Mehrere Filterkriterien verwenden

In diesem Abschnitt werde ich Ihnen zeigen, wie Sie mehrere Filterkriterien gleichzeitig verwenden und miteinander kombinieren können. Der einfachste Fall ist, dass Sie das Ergebnis einer Abfrage dazu verwenden, in einer zweiten Abfrage einen weiteren Filter anzuwenden.

Eine andere Abfrage als Datensatzquelle verwenden

Zunächst erstellen wir den ersten Filter für alle männlichen Mitarbeiter, den Sie bereits kennen:

1. Öffnen Sie die Datenbank *O3_Abfragen\3_Fluege.accdb*.
2. Erstellen Sie eine neue Abfrage in der Entwurfsansicht.
3. Fügen Sie die Tabelle *tblMitarbeiter* hinzu.
4. Ziehen Sie das Feld mit dem Sternchen per Drag & Drop in den unteren Bereich (alternativ: Doppelklick auf das Sternchen).
5. Ziehen Sie das Feld »Geschlecht« per Drag & Drop in den unteren Bereich.
6. Entfernen Sie für das Feld »Geschlecht« das Häkchen bei ANZEIGEN.
7. Tragen Sie für das Feld »Geschlecht« in der Zeile KRITERIEN "M" ein.
8. Speichern Sie die Abfrage unter dem Namen *qryMitarbeiterMaennlich*. Anschließend erstellen Sie die zweite Abfrage, die auf der ersten basiert:
9. Erstellen Sie eine neue Abfrage in der Entwurfsansicht.
10. Fügen Sie die Abfrage *qryMitarbeiterMaennlich* hinzu.

Sie finden alle bereits gespeicherten Abfragen im Dialog TABELLE ANZEIGEN unter dem Registerblatt ABFRAGEN.

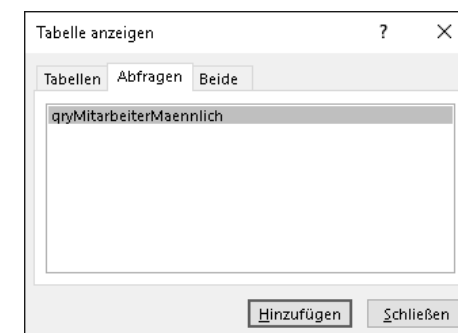


Abbildung 3.12 Als Datensatzquelle für eine Abfrage können Sie sowohl eine Tabelle als auch eine andere, bereits bestehende Abfrage verwenden.

11. Ziehen Sie das Feld mit dem Sternchen per Drag & Drop in den unteren Bereich.
12. Ziehen Sie das Feld »Geburtsdatum« per Drag & Drop in den unteren Bereich.
13. Entfernen Sie für das Feld »Geburtsdatum« das Häkchen bei ANZEIGEN.
14. Tragen Sie für das Feld »Geburtsdatum« in der Zeile KRITERIEN `<DatAdd("jjjj";-40;Jetzt())` ein. Mit dieser Formel werden nur diejenigen Mitarbeiter durch den Filter gelassen, deren Geburtsdatum mindestens 40 Jahre vor dem aktuellen Datum liegt.
15. Speichern Sie die Abfrage unter dem Namen *qryMitarbeiterMaennlich_AelterAls40Jahre*, und schalten Sie in die Datenblattansicht um.

Sie finden dieses Beispiel in den Materialien zum Buch in der Datenbank *03_Abfragen\3.1.4_Mehrere_Auswahlkriterien.accdb*. Nach diesem Schema können Sie eine ganze Kette von Filterkriterien über mehrere Abfragen hinweg einrichten. Angezeigt werden nur diejenigen Datensätze, die durch alle Filter durchgelassen werden.

Mehrere Filterkriterien innerhalb einer Abfrage kombinieren

Mehrere Filterkriterien können auch innerhalb ein und derselben Abfrage eingetragen werden. Besonders schön ist, dass Sie die einzelnen Kriterien logisch miteinander verbinden können. Grundsätzlich gilt:

- ▶ Alle Filterkriterien, die innerhalb einer KRITERIEN-Zeile stehen, gelten gleichzeitig (UND-Verknüpfung).
- ▶ Die verschiedenen KRITERIEN-Zeilen lassen unabhängig voneinander Datensätze durch den Filter durch (ODER-Verknüpfung).
- ▶ Zusätzlich können Sie in einer Zelle für ein Filterkriterium mehrere Ausdrücke mit den Schlüsselwörtern Und, Oder, ExOder (*exklusives Oder*) und Nicht miteinander verknüpfen. Dabei können Sie mit Hilfe von runden Klammern die Reihenfolge der Operatoren festlegen.

Schauen wir uns als Beispiel einmal die Abfrage *qryProjektAktiv* an, die Sie in den Materialien zum Buch in der Datenbank *03_Abfragen\3.1.4_Mehrere_Auswahlkriterien.accdb* finden.

Feld:	ProjektID	ProjektAbkuerzung	Beginn	Ende
Tabelle:	tblProjekt	tblProjekt	tblProjekt	tblProjekt
Sortierung:				
Anzeigen:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kriterien:			<=Jetzt()	Ist Null
oder:				

Abbildung 3.13 Ein Filter für alle aktiven Projekte, der zwei Filterkriterien enthält

Sie erkennen, dass alle Filterkriterien innerhalb einer Zeile stehen. Sowohl der Filter für das Feld »Beginn« als auch der für das Feld »Ende« gelten gleichermaßen (UND-Verknüpfung).

1. Der Filter für das Feld »Beginn«

Die Formel

<=Jetzt()

lässt nur diejenigen Datensätze durch den Filter, in denen das Datum des Projektbeginns in der Vergangenheit liegt.

2. Der Filter für das Feld »Ende«

Das Kriterium

Ist Null

lässt nur solche Datensätze zu, für die das Projektende noch nicht feststeht.

Da beide Filterkriterien in einer Zeile stehen, werden nur diejenigen Projekte in der Datenblattansicht angezeigt, deren Projektbeginn in der Vergangenheit liegt *und* deren Projektende noch nicht feststeht.

Lassen Sie uns das Beispiel noch etwas verfeinern. Glücklicherweise kann man für die meisten Projekte das Ende früher oder später absehen und in die Tabelle *tblProjekt* eintragen. Dies müssen wir beim Filterkriterium für das Feld »Ende« berücksichtigen. Aktive Projekte sind diejenigen Projekte, deren Ende noch nicht feststeht

Ist Null

oder deren Ende zum gegenwärtigen Zeitpunkt in der Zukunft liegt

>Jetzt()

Beide Filterkriterien verknüpfen wir mit dem Schlüsselwort Oder und tragen den gesamten Ausdruck in die entsprechende Zelle ein. Gleichzeitig muss nach wie vor das Filterkriterium für das Feld »Beginn« gelten. Achten Sie daher bitte darauf, dass alle Filterkriterien innerhalb einer Zeile stehen! Unsere Abfrage sieht jetzt so aus:

Feld:	ProjektID	ProjektAbkuerzung	Beginn	Ende
Tabelle:	tblProjekt	tblProjekt	tblProjekt	tblProjekt
Sortierung:				
Anzeigen:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kriterien:			<=Jetzt()	Ist Null Oder >Jetzt()
oder:				

Abbildung 3.14 In der Abfrage »qryProjektAktiv2« werden drei Filterkriterien miteinander verknüpft.

Diese Abfrage führt zur gewünschten Liste aller aktiven Projekte. Von der Logik her ist diese Anordnung der Filterkriterien übrigens gleichwertig mit dieser Version (Abbildung 3.15).

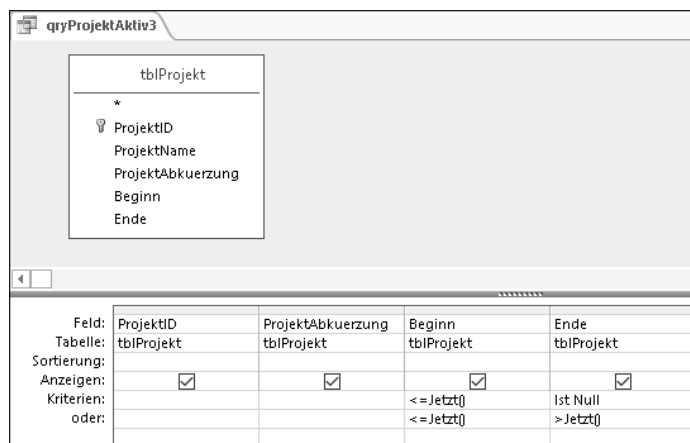


Abbildung 3.15 Diese Verknüpfung von vier Filterkriterien (»qryProjektAktiv3«) ist logisch gleichwertig mit den Filterkriterien in »qryProjektAktiv2«.

Optionale Felder benötigen bei den Filterkriterien besondere Aufmerksamkeit

Wie Sie auch in diesem Beispiel sehen, müssen NULL-Werte häufig gesondert berücksichtigt werden. Denken Sie bei optionalen Feldern immer daran, dass der Feldinhalt auch NULL sein kann. Mit den Operatoren =, >, <, >=, <= und <> gehen Ihnen alle Datensätze mit NULL-Werten verloren! Verwenden Sie die Operatoren Ist Null und Ist Nicht Null, um auf NULL-Werte zu prüfen.

Optionale Felder erzeugen genau diesen zusätzlichen Aufwand bei Filtern und Fallunterscheidungen. Daher empfehle ich Ihnen, wo immer dies sinnvoll ist, NOT-NULL-Felder einzusetzen (vergleiche Abschnitt 2.3.12, »Feldeinschränkungen«).

3.1.5 Datensätze sortieren

Ich habe Ihnen bereits in Abschnitt 3.1.1, »Vertikales Filtern – Felder auswählen«, gezeigt, wie Sie die Reihenfolge der Felder in der Entwurfsansicht einer Abfrage festlegen können. In diesem Abschnitt beschäftigen wir uns nun mit der Reihenfolge der Datensätze, der *Sortierung*.

Aufsteigende Sortierung

In der Entwurfsansicht einer Abfrage ist Ihnen vielleicht schon die Zeile SORTIERUNG aufgefallen. Für die Felder, nach deren Inhalt die Datensätze sortiert werden sollen, können Sie AUFSTEIGEND oder ABSTEIGEND wählen. Ein paar Beispiele dazu finden Sie in den Materialien zum Buch in der Datenbank *03_Abfragen\3.1.5_Datensaetze_sortieren.accdb*.

So erstellen Sie eine Liste aller Mitarbeiter, sortiert nach Nachname und Vorname:

1. Erstellen Sie eine neue Abfrage in der Entwurfsansicht.
2. Fügen Sie die Tabelle *tblMitarbeiter* hinzu.
3. Ziehen Sie das Feld »MitarbeiterID« per Drag & Drop in den unteren Bereich (alternativ: Doppelklick auf das Feld).
4. Ziehen Sie die Felder »Nachname« und »Vorname« in den unteren Bereich. Wählen Sie für beide Felder unter SORTIERUNG den Eintrag AUFSTEIGEND aus.
5. Ziehen Sie das Feld »Geschlecht« in den unteren Bereich.
6. Speichern Sie die Abfrage unter dem Namen *qryMitarbeiterSortiertNachNachnameVorname*.

Wenn Sie in die Datenblattansicht wechseln, erhalten Sie die sortierte Liste aller Mitarbeiter. Die Anordnung der Felder ist wichtig: Es wird zuerst nach Nachname und dann nach Vorname sortiert.

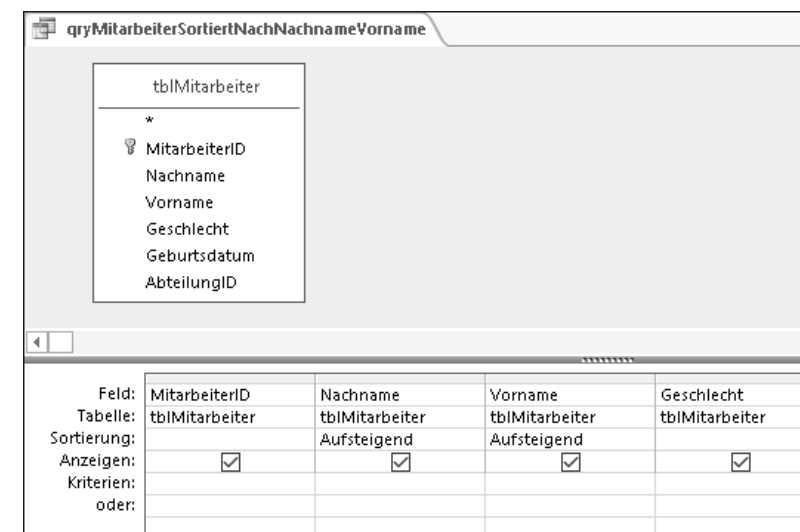


Abbildung 3.16 In der Zeile »Sortierung« können Sie einstellen, nach welchen Feldern die Datensätze sortiert werden sollen.

Sie möchten zuerst nach Vorname und dann nach Nachname sortieren? Das ist natürlich auch möglich.

1. Wechseln Sie wieder in die Entwurfsansicht der Abfrage.
2. Verschieben Sie nun im unteren Bereich das Feld »Nachname« nach rechts neben »Vorname«.

Wichtig ist, in welcher Reihenfolge die Felder, die Sie sortieren möchten, nebeneinander angeordnet sind, in diesem Fall also erst »Vorname«, dann »Nachname«.

Die Reihenfolge der Sortierung unabhängig von der Reihenfolge der Felder einstellen

Die Sortierung stimmt so weit. Aber was ist, wenn ich unbedingt in der zweiten Spalte das Feld »Nachname« und erst in der dritten Spalte das Feld »Vorname« sehen möchte? Um dieses Ziel zu erreichen, müssen wir einen kleinen Trick nutzen:

1. Wechseln Sie wieder in die Entwurfsansicht der Abfrage.
2. Ziehen Sie das Feld »Nachname« noch einmal per Drag & Drop in den unteren Bereich, und zwar links neben »Vorname«. Stellen Sie für dieses Feld »Nachname« *keine Sortierung* ein.
3. Entfernen Sie das Häkchen in der Zeile ANZEIGEN für das andere Feld »Nachname«.

Den letzten Schritt sollten Sie nicht vergessen, denn sonst erscheint das Feld »Nachname« doppelt in der Datenblattansicht. An dieser Stelle zeigt es sich, wie nützlich Felder in Abfragen sind, die nicht angezeigt werden. Sie können nicht angezeigte Felder zum Sortieren und zum Filtern verwenden! Sie finden diese Abfrage in der Datenbank in den Materialien zum Buch unter dem Namen *qryMitarbeiterSortiertNachVornameNachname*.

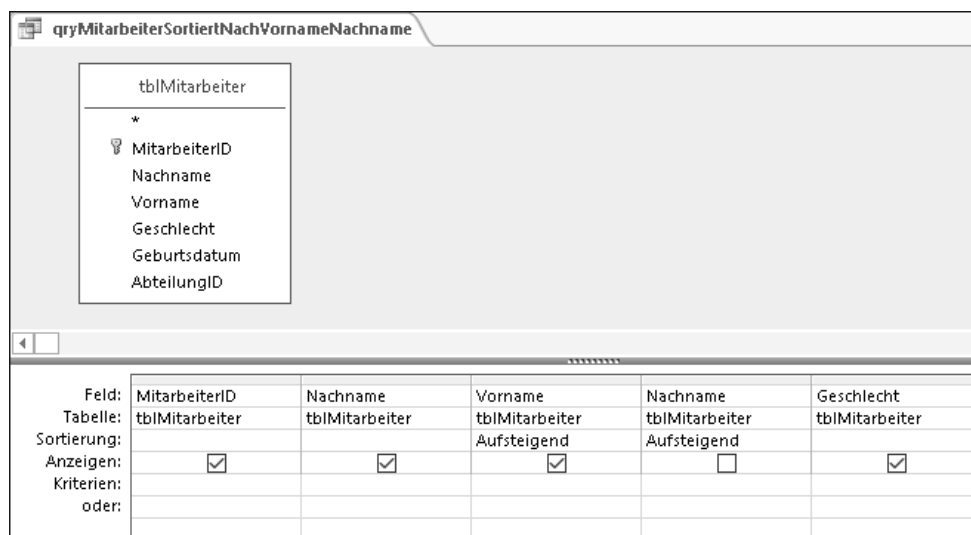


Abbildung 3.17 Ausgeblendete Felder sind ein nützlicher Trick beim Filtern und Sortieren.

Absteigende Sortierung

Absteigende Sortierung ist eigentlich selbsterklärend. Schauen Sie sich bitte einmal die Abfrage *qryMitarbeiterAbsteigendSortiert* in der Datenbank in den Materialien zum Buch an. Dies ist die absteigend sortierte Mitarbeiterliste.

MitarbeiterID	Nachname	Vorname	Geschlecht
10	Zimmermann	Arno	M
9	Wilke	Margot	W
8	Wagner	Michael	M
7	Semrau	Gabriele	W
6	Schulz	Barbara	W
5	Schreiber	Alois	M
4	Rathke	Ramona	W
3	Leuschner	Doris	W
2	Hachmann	Eva	W
1	Guntermann	Hildegard	W
*	(Neu)		

Abbildung 3.18 Die Mitarbeiterliste in absteigender Sortierung

Die Sortierung erkennt automatisch den Felddatentyp

Access hat eine sehr zuverlässige Sortierfunktion, die den Felddatentyp berücksichtigt. Besonders praktisch ist das bei Datums- und Uhrzeitangaben. Es wird immer von alt nach neu (bzw. umgekehrt) sortiert – und zwar unbeeinflusst von der Darstellung am Bildschirm, der Formatierung. Sie kennen bestimmt den Altraum auf dem Dateisystem: Sie haben gerade Ihre letzten Urlaubsbilder mit Datum und Uhrzeit im Dateinamen umbenannt ... Wer das einmal mit deutschem Datumsformat ausprobiert hat, der weiß, wovon ich spreche: Die Dateien erscheinen im Windows-Explorer nicht mehr in chronologischer Reihenfolge.

01.09.2019 nach dem Urlaub.jpeg

09.08.2019 am Strand.jpeg

10.08.2019 in den Bergen.jpeg

13.07.2019 vor der Abreise.jpeg

In Access kann Ihnen das nicht passieren! Access sortiert immer korrekt. Beispiele dazu finden Sie in den Materialien zum Buch in den Abfragen *qryMitarbeiterSortiertNachGeburtsdatum* und *qryFlugSortiertNachFluggesellschaft*.

Felddatentyp	aufsteigend	absteigend
KURZER TEXT	A–Z	Z–A
ZAHL	klein–groß	groß–klein
WÄHRUNG	wenig–viel	viel–wenig
DATUM/UHRZEIT	alt–neu	neu–alt

Tabelle 3.2 Access berücksichtigt bei der Sortierung den Felddatentyp.

Sortieren und Filtern in einer Abfrage

Ich hatte eingangs erwähnt, dass Sie in Abfragen gleichzeitig filtern und sortieren können. Dazu möchte ich Ihnen zum Abschluss eine recht simple, aber nützliche Abfrage mitgeben. Schauen Sie sich bitte einmal die Abfrage *qryFlugbuchung_Die_teuersten_drei_Flugbuchungen* an:

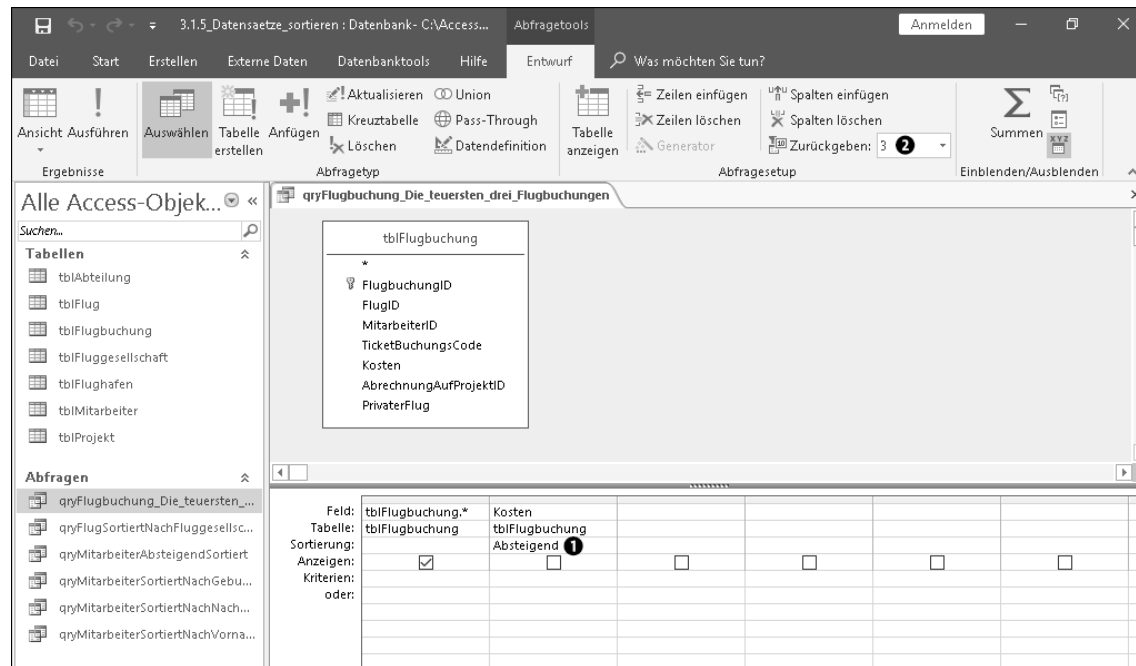


Abbildung 3.19 In einer Abfrage können Sie gleichzeitig sortieren ❶ und filtern. Der Filter versteckt sich in dieser Abfrage im Feld »Zurückgeben« ❷.

Diese Abfrage verwirklicht gleichzeitig eine Sortierung (die teuersten Flugbuchungen zuerst, daher absteigende Sortierung ❶) und horizontales Filtern (bitte nur die ersten drei; hier realisiert über das Feld ZURÜCKGEBEN ❷).

Auf einen Blick

1	Einleitung	21
2	Access als Datenbank: Tabellen	37
3	Daten filtern, sortieren und zusammenfassen: Abfragen	219
4	Access und die Verbindungen zur Außenwelt	329
5	SQL – die Programmiersprache für Datenbanken	509
6	Formulare	559
7	Berichte	745
8	Einfache Programmierung mit Makros	815
9	Visual Basic for Applications (VBA), die Programmiersprache für Microsoft-Office-Anwendungen	851
10	Data-Access-Objects-(DAO-)Klassenbibliothek	961
11	Anwendungsprogramme mit Access erstellen	991

Inhalt

Vorwort	17
1 Einleitung	21
1.1 Was ist Access?	21
1.1.1 Relationale Datenbank	23
1.1.2 Formulare zum Bearbeiten von Daten	24
1.1.3 Berichte zur Auswertung und zum übersichtlichen Ausdrucken von Daten	25
1.1.4 Programmierung: Optional, zum Beispiel zum Lösen komplexerer Probleme	26
1.1.5 Schnelle Entwicklung kompletter Anwendungsprogramme (»rapid development«)	26
1.2 Wie kann eine Datenbank meine Arbeitsprozesse unterstützen?	27
1.2.1 Beispiel einer einfachen Desktop-Datenbank: Mitgliederverwaltung eines Vereins	28
1.2.2 Beispiel einer komplexen Access-Desktop-Datenbank: Erfassung von Mitarbeitern, Flügen und Flugbuchungen	32
1.2.3 Access Web Apps in SharePoint gibt es nicht mehr	36
2 Access als Datenbank: Tabellen	37
2.1 Ein Schnelleinstieg in relationale Datenbanken	37
2.1.1 Eine neue Desktop-Datenbank mit Access erstellen	38
2.1.2 Eine Tabelle in Access erstellen	40
2.1.3 Die Tabelle mit Inhalt füllen	42
2.2 Datenbankdesign	43
2.2.1 Die Welt in Tabellen abbilden	44
2.2.2 Namenskonventionen	46
2.2.3 Der Primärschlüssel	51
2.2.4 Die Tabellenstruktur festlegen	53
2.3 Erstellen von Tabellen	54
2.3.1 Die Entwurfsansicht	54
2.3.2 Felddatentyp »Kurzer Text«	60

2.3.3	Felddatentyp »Langer Text«	61
2.3.4	Felddatentyp »Zahl«	63
2.3.5	Felddatentyp »AutoWert«	68
2.3.6	Felddatentyp »Datum/Uhrzeit«	69
2.3.7	Felddatentyp »Währung«	73
2.3.8	Felddatentyp »Ja/Nein«	74
2.3.9	Felddatentyp »OLE-Objekt«	75
2.3.10	Andere Felddatentypen	75
2.3.11	Berechnete Felder	76
2.3.12	Feldeinschränkungen	78
2.3.13	Tabelleneinschränkungen	88
2.3.14	Der Nachschlage-Assistent	94
2.4	Ändern von Tabellen	94
2.4.1	Hinzufügen von Feldern	94
2.4.2	Verschieben von Feldern	96
2.4.3	Löschen von Feldern	96
2.4.4	Ändern des Felddatentyps	98
2.4.5	Ändern von Feld- und Tabelleneinschränkungen	99
2.5	Daten in Tabellen	100
2.5.1	Daten eingeben, ändern und löschen	100
2.5.2	Navigationsschaltflächen	105
2.5.3	Die Spaltenbreite eines Feldes anpassen	105
2.5.4	Felder sortieren	106
2.5.5	Felder ein- und ausblenden	106
2.5.6	Felder fixieren	107
2.5.7	Datensätze sortieren	108
2.5.8	Suchfunktionen	111
2.5.9	Datensätze filtern	114
2.6	Formatierungen in Tabellen	123
2.6.1	Format	124
2.6.2	Eingabeformat	138
2.6.3	Beschreibung und Beschriftung	147
2.6.4	Nachschlagen	147
2.7	Beziehungen zwischen Tabellen	150
2.7.1	1:n-Beziehung	150
2.7.2	1:1-Beziehung	153
2.7.3	m:n-Beziehung	156
2.7.4	Referentielle Integrität	159
2.7.5	Übersicht schaffen mit Datenbankdiagrammen	165
2.7.6	Typische Muster von Tabellenbeziehungen	172

2.8	Durch die Normalformen das Datenbankdesign verbessern	180
2.8.1	Die nicht normalisierte Form	181
2.8.2	Die erste Normalform (1NF)	181
2.8.3	Die zweite Normalform (2NF)	183
2.8.4	Die dritte Normalform (3NF)	189
2.8.5	Die Boyce-Codd-Normalform (BCNF)	193
2.8.6	Die vierte Normalform (4NF)	196
2.8.7	Weitere Normalformen	199
2.8.8	Zweckmäßiges Datenbankdesign	201
2.9	Indizes	205
2.9.1	Automatisch generierte Tabellenindizes	205
2.9.2	Eindeutiger Schlüssel (Alternativschlüssel)	208
2.9.3	Erstellen eines Indexes	211
2.9.4	Löschen eines Indexes	214
2.10	Besondere Tabellen	215
2.10.1	Versteckte Datenbankobjekte	215
2.10.2	Systemtabellen (»MSys«) und Usertabellen (»USys«)	216
2.10.3	Das Verzeichnis aller Datenbankobjekte: »MSysObjects«	217
3	Daten filtern, sortieren und zusammenfassen: Abfragen	219
3.1	Einfache Abfragen	219
3.1.1	Vertikales Filtern – Felder auswählen	221
3.1.2	Horizontales Filtern – Datensätze auswählen	226
3.1.3	Umgang mit NULL-Werten	235
3.1.4	Mehrere Filterkriterien verwenden	236
3.1.5	Datensätze sortieren	240
3.2	Auswerten von Daten eines Datensatzes: Berechnete Felder	245
3.2.1	Einfache Operationen für Zahlenfelder	246
3.2.2	Aneinanderreihen von Textfeldern	249
3.2.3	Grundsätzliches zu Funktionen	249
3.2.4	Funktionen für Textfelder	251
3.2.5	Funktionen für Zahlenfelder	252
3.2.6	Funktionen für Datumsfelder	254
3.2.7	Konvertieren des Felddatentyps	256
3.2.8	Einfache Fallunterscheidungen	259
3.2.9	Parameter in Abfragen	262

3.3 Zusammenfassen von Datensätzen: Gruppierung und Aggregieren	266
3.3.1 Datensätze zu Gruppen zusammenfassen	268
3.3.2 Die Anzahl der Datensätze pro Gruppe	269
3.3.3 Alle Zahlen einer Gruppe summieren	271
3.3.4 Minimal- und Maximalwert aller Zahlen einer Gruppe	272
3.3.5 Mittelwert, Varianz und Standardabweichung aller Zahlen einer Gruppe	273
3.3.6 Datensätze vor dem Zusammenfassen filtern – Filter auf die Rohdaten	274
3.3.7 Datensätze nach dem Zusammenfassen filtern – Filter auf das Ergebnis	276
3.3.8 Kreuztabellenabfrage	278
3.4 Abfragen von Daten aus mehreren Tabellen	282
3.4.1 Unterdatenblätter	283
3.4.2 Passende Datensätze zusammenbringen: Abfragen mit INNER-JOIN-Verknüpfung	285
3.4.3 Jeder Datensatz mit jedem gepaart: Das kartesische Produkt	289
3.4.4 Wenn vorhanden, dann nur passende Datensätze zusammen: Abfragen mit OUTER-JOIN-Verknüpfung	291
3.4.5 Abfragen mit mehr als zwei Tabellen	295
3.5 Aktionsabfragen	299
3.5.1 Anfügeabfragen	301
3.5.2 Löschartabfragen	305
3.5.3 Aktualisierungsabfragen	308
3.6 Workshops zum Normalisieren von Daten	315
3.6.1 Die Haupttabelle einer 1:n-Beziehung füllen	315
3.6.2 Die Suche nach Inkonsistenzen	320
3.6.3 Inkonsistenzen beheben	323
3.6.4 Dublettensuche	326
4 Access und die Verbindungen zur Außenwelt	329
4.1 Wann braucht man Datenverbindungen?	329
4.1.1 Eine Access-Datenbank als Insel-Lösung verwenden	331
4.1.2 Import und Export von Daten	333
4.1.3 Verbundene Datenquellen	336
4.1.4 Verteilte Datenquellen – »Datawarehouse« und »Datamining«	338
4.2 Export von Daten	339
4.2.1 Export über die Zwischenablage	339
4.2.2 Export in eine andere Access-Datenbank	342
4.2.3 Export nach Excel	343

4.2.4 Export nach Word	345
4.2.5 Export als PDF- oder XPS-Datei	347
4.2.6 Export in eine Textdatei	349
4.2.7 Export in eine HTML-Datei	355
4.2.8 Export in eine dBASE-Datei	357
4.2.9 Export in eine XML-Datei	360
4.2.10 Daten als E-Mail versenden	365
4.2.11 Export als SharePoint-Liste	366
4.2.12 Gespeicherte Exporte	368
4.3 Zugriff auf eine Access-Desktop-Datenbank aus anderen Programmen	372
4.3.1 Öffnen einer Access-Tabelle in Excel	372
4.3.2 Serienbriefe in Word erstellen	377
4.3.3 Zugriff auf eine Access-Datenbank vom Microsoft SQL Server aus	382
4.4 Import von Daten	389
4.4.1 Import aus einer anderen Access-Datenbank	390
4.4.2 Importfehler	392
4.4.3 Import eines Outlook-Ordners	397
4.4.4 Import aus einer XML-Datei	401
4.4.5 Import aus einer dBASE-Datei	404
4.4.6 Daten per E-Mail sammeln	406
4.4.7 Import einer SharePoint-Liste	406
4.4.8 Datendienste nutzen	408
4.4.9 Gespeicherte Importe	408
4.5 Workshops zum Import von Daten	409
4.5.1 Workshop: Einmaliger Import von Daten	410
4.5.2 Workshop: Regelmäßiger Datenabgleich per Import	415
4.5.3 Workshop: Import einer Tabelle mit fehlenden Daten	426
4.6 Anbindung externer Datenquellen: Verknüpfen von Tabellen	458
4.6.1 Eine Tabelle aus einer anderen Access-Datenbank verknüpfen	458
4.6.2 Trennung von Daten und Anwendung	460
4.6.3 Aktualisieren von Tabellenverknüpfungen	463
4.6.4 Andere Dateiformate verknüpfen	465
4.6.5 Eine SharePoint-Liste verknüpfen	467
4.7 Was sind Server-Datenbanken?	468
4.7.1 Vor- und Nachteile gegenüber Access	469
4.7.2 Häufig eingesetzte Datenbank-Server	472
4.7.3 Eine Tabelle über die ODBC-Schnittstelle verknüpfen	473
4.7.4 Optimierung der Client-Server-Anwendung	484
4.7.5 Access-Projekte	485

4.8 Workshops zum SQL Server	487
4.8.1 Workshop: Tabellen im SQL Server erstellen und in Access nutzen	487
4.8.2 Workshop: Datenbankdesign auf dem SQL Server	497
4.8.3 Workshop: Eine bestehende Access-Datenbank auf den SQL Server migrieren	503
5 SQL – die Programmiersprache für Datenbanken	509
5.1 Warum SQL lernen?	509
5.1.1 Voraussetzungen für dieses Kapitel	509
5.1.2 Der grafische Abfrage-Editor und SQL	510
5.2 Daten abfragen: SELECT	510
5.2.1 Abfragen von Daten aus einer Tabelle	511
5.2.2 Datensätze filtern	516
5.2.3 Datensätze sortieren	519
5.2.4 Parameter in Abfragen	520
5.2.5 Berechnete Felder	521
5.2.6 Abfragen von Daten aus mehreren Tabellen	521
5.3 Gruppierung und Aggregieren	528
5.3.1 GROUP BY	528
5.3.2 Aggregatfunktionen	529
5.3.3 HAVING	531
5.4 Unterabfragen	532
5.4.1 In der SELECT-Klausel anstelle eines Tabellenfeldes	533
5.4.2 In der FROM-Klausel anstelle einer Tabelle	536
5.4.3 In der WHERE-Klausel	537
5.4.4 In der ORDER-BY-Klausel anstelle eines Tabellenfeldes	542
5.5 Aktionsabfragen: Data Manipulation Language (DML)	543
5.5.1 INSERT INTO	544
5.5.2 SELECT ... INTO	546
5.5.3 DELETE	547
5.5.4 UPDATE	549
5.6 Datenbankobjekte verändern: Data Definition Language (DDL)	550
5.6.1 CREATE TABLE	551
5.6.2 ALTER TABLE	552
5.6.3 DROP TABLE	553
5.6.4 CREATE INDEX	553
5.6.5 DROP INDEX	554

5.7 Besondere Abfragen	554
5.7.1 UNION-Abfragen	554
5.7.2 Pass-through-Abfragen	557
6 Formulare	559
6.1 Formulare entwerfen	559
6.1.1 Ein einfaches Formular in der Entwurfsansicht erstellen	559
6.1.2 Datensatzquelle	560
6.1.3 Einen Datensatz in der Formularansicht bearbeiten	562
6.1.4 Formulare als Registerblätter oder als Fenster	562
6.1.5 Welche Ansichten von Formularen gibt es?	564
6.1.6 Navigationsschaltflächen	565
6.1.7 Datensatzmarkierer	569
6.2 Arbeiten mit Steuerelementen	570
6.2.1 Hinzufügen und bearbeiten	570
6.2.2 Anpassen von Größe und Abstand	580
6.2.3 Ausrichten	583
6.2.4 Visuelles Erscheinungsbild	584
6.2.5 Aktivieren und deaktivieren	587
6.2.6 Formatierungen	590
6.2.7 Bedingte Formatierungen	592
6.2.8 Datenbalken	595
6.2.9 Reihenfolge der Aktivierung	596
6.2.10 Gruppierung von Steuerelementen	597
6.3 Die verschiedenen Steuerelemente	597
6.3.1 Welche Steuerelemente gibt es?	598
6.3.2 Steuerelemente für Text	599
6.3.3 Auswahl Ja/Nein	603
6.3.4 Auswahl aus verschiedenen Optionen	606
6.3.5 Auswahl aus einer Liste	619
6.3.6 Grafik-Steuerelemente	626
6.3.7 Diagramme generieren	647
6.3.8 Internetfunktionen	655
6.3.9 Andere Steuerelemente	659
6.4 Arbeiten mit Layouts in Formularen	666
6.4.1 Layout- und Entwurfsansicht	667
6.4.2 Layout hinzufügen	668
6.4.3 Den Layoutbereich anpassen	670

6.4.4	Layout entfernen	672
6.4.5	Layout ähnlich einer Tabelle: Formularkopf und Formularfuß	673
6.4.6	Abstand und Gitternetzlinien	674
6.4.7	Steuerelemente verankern	675
6.5	Formulartypen	679
6.5.1	Das gebundene Standardformular	680
6.5.2	Das ungebundene Standardformular	680
6.5.3	Das Endlosformular	681
6.5.4	Das geteilte Formular	684
6.5.5	Das einfache Popup-Formular	686
6.5.6	Das modale Popup-Formular	687
6.5.7	Nicht mehr unterstützte Formulartypen	689
6.6	Arbeiten mit mehreren Formularen	689
6.6.1	Zugriff auf Daten in anderen Formularen	690
6.6.2	Unterformulare	701
6.6.3	Synchronisieren von Unterformularen	704
6.6.4	Das Navigationssteuerelement	709
6.7	Formulare aus einer Vorlage erstellen	712
6.7.1	Formularvorlagen	712
6.7.2	Anwendungsparts	713
6.8	Workshops zu Formularen	715
6.8.1	Workshop: Formulare und Unterformulare im Zusammenspiel	715
6.8.2	Workshop: Eine individuelle Eingabe-Dialogbox	723
6.8.3	Workshop: Platzreservierung	736
7	Berichte	745
7.1	Berichte entwerfen	745
7.1.1	Einen einfachen Bericht in der Entwurfsansicht erstellen	745
7.1.2	Datensatzquelle	747
7.1.3	Steuerelemente in Berichten	749
7.1.4	Unterberichte	754
7.2	Arbeiten mit Layouts in Berichten	759
7.2.1	Layout hinzufügen	759
7.2.2	Unterschiede zu Formularen	760
7.3	Ansichten eines Berichts	760
7.3.1	Seitenansicht	762
7.3.2	Direktes Drucken eines Berichts	764

7.3.3	Berichtsansicht	765
7.3.4	Einen Bericht als PDF- oder XPS-Datei exportieren	766
7.4	Bereiche in Berichten	767
7.4.1	Kopf- und Fußbereich	767
7.4.2	Ebenen zur Gruppierung	771
7.4.3	Seitenumbrüche setzen	785
7.4.4	Gruppen zusammenhalten und Abstände richtig setzen	786
7.4.5	Wiederholende Einträge ausblenden	788
7.4.6	Laufende Summen und Gesamtsummen	790
7.5	Berichte mit mehreren Spalten	792
7.5.1	Etiketten mit dem Assistenten erstellen	792
7.5.2	Mehrere Spalten einrichten	793
7.5.3	Der Seitenkopf für mehrere Spalten	797
7.6	Workshops zu Berichten	800
7.6.1	Workshop: Summen und Prozentsätze	800
7.6.2	Workshop: Filterformulare für Berichte	807
7.6.3	Workshop: Bericht in eine HTML-Datei exportieren	811
8	Einfache Programmierung mit Makros	815
8.1	Makros	815
8.1.1	Aktionen	816
8.1.2	Unsichere Aktionen und vertrauenswürdige Datenbanken	820
8.1.3	Kommentare im Makro	827
8.1.4	Fallunterscheidungen	827
8.1.5	Gruppieren und Untermakros	832
8.2	Auf Ereignisse reagieren	834
8.2.1	Ereignisse von Formularen	835
8.2.2	Ereignisse von Steuerelementen	838
8.2.3	Mit dem Zeitgeber arbeiten (Timer)	840
8.2.4	Ereignisse eines Berichts	840
8.2.5	Eingebettete Makros	842
8.3	Besondere Makros	843
8.3.1	Datenmakros (Trigger)	843
8.3.2	»AutoExec« startet beim Öffnen der Datenbank	848
8.3.3	Tastaturkürzel in »AutoKeys«	849

9	Visual Basic for Applications (VBA), die Programmiersprache für Microsoft-Office- Anwendungen	851
9.1	Warum VBA lernen?	851
9.1.1	Voraussetzungen für dieses Kapitel	852
9.1.2	Die Zukunft von VBA	852
9.2	Formulare bereichern	853
9.2.1	Auf Ereignisse reagieren	853
9.2.2	Kommentare im Code	856
9.2.3	Berechnungen und Formeln	856
9.2.4	Fallunterscheidungen	857
9.2.5	Meldungen am Bildschirm anzeigen: »MsgBox«	859
9.2.6	Befehle, die über mehrere Zeilen reichen	862
9.3	Die integrierte Entwicklungsumgebung	863
9.3.1	Ansicht im Codefenster	863
9.3.2	Übersicht halten mit dem Objektkatalog und dem Projekt-Explorer	866
9.3.3	VBA-Code im Direktbereich ausführen	869
9.3.4	Programme starten	870
9.3.5	Haltepunkte und Debug-Modus	871
9.4	Daten im Speicher verarbeiten	873
9.4.1	Variablen	873
9.4.2	Konstanten	881
9.4.3	Eingaben vom Anwender abfragen: »InputBox()«	882
9.4.4	Statische Variablen	884
9.4.5	Arrays	885
9.4.6	Dynamische Arrays	886
9.5	Kontrollstrukturen: Verzweigungen und Wiederholungen	887
9.5.1	Bedingungen und Fallunterscheidungen	888
9.5.2	Fallauswahl	888
9.5.3	Schleifen	889
9.5.4	Iteration	893
9.5.5	Sprungbefehle und Sprungmarken	894
9.6	Module	895
9.6.1	Ein Modul erstellen und bearbeiten	895
9.6.2	Prozeduren	896
9.6.3	Funktionen	899

9.6.4	Parameter	907
9.6.5	Gültigkeitsbereiche	917
9.6.6	Variablen auf Modulebene und globale Variablen	921
9.7	Klassenmodule	923
9.7.1	Ein eigenständiges Klassenmodul erstellen und bearbeiten	925
9.7.2	Objektvariablen	928
9.7.3	Eigenschaften (Properties)	935
9.7.4	Der Befehl »With«	943
9.7.5	Auflistungen (Collections)	945
9.8	Fehlerbehandlung	951
9.8.1	Auf Laufzeitfehler reagieren	952
9.8.2	Die Behandlung von Fehlern	957
9.8.3	Eigene Laufzeitfehler auslösen	958
9.8.4	Zentrale Fehlerbehandlung	958
10	Data-Access-Objects-(DAO-)Klassenbibliothek	961
10.1	Daten abfragen: »Recordset«-Objekte	962
10.1.1	Lesend auf eine Tabelle zugreifen	962
10.1.2	Einen Datensatz suchen	966
10.1.3	Domänenfunktionen – Datenabfrage in einer Zeile	968
10.1.4	Datensätze verändern	970
10.1.5	Abfragen erstellen und als Datensatzquelle nutzen	971
10.2	Aktionsabfragen ausführen	974
10.2.1	Bestehende Aktionsabfragen aufrufen	974
10.2.2	SQL-Code direkt ausführen	976
10.3	Transaktionen	976
10.3.1	Eine Transaktion starten und abschließen	977
10.3.2	Änderungen doch nicht durchführen: Rollback	979
10.4	Andere Klassenbibliotheken	980
10.4.1	Referenzen definieren	981
10.4.2	Word-Dokumente bearbeiten	982
10.4.3	Daten aus einer Excel-Tabelle gezielt auslesen	984
10.4.4	ActiveX Data Objects (ADO)	985
10.4.5	Zugriff auf das Windows-Betriebssystem	987

11	Anwendungsprogramme mit Access erstellen	991
11.1	Wichtige Elemente eines Anwendungsprogramms	992
11.1.1	Start der Anwendung	993
11.1.2	Formen des Hauptmenüs	994
11.1.3	Menüband	996
11.2	Zugriffsberechtigungen	1003
11.2.1	Benutzer, Gruppen und Zugriffsberechtigungen in ».mdb«-Dateien	1003
11.2.2	Verschlüsseln der Datenbank	1011
11.2.3	Erstellen einer ».accde«-Datei	1013
11.3	Ohne Access mit Datenbank-Dateien arbeiten: Die Access Runtime	1014
11.3.1	Installation der Access Runtime	1014
11.3.2	Workshop: Eine Datenbank runtime-fähig machen	1017
11.3.3	Workshop: Anwendungsprogramm mit Backend-Datenbank	1023
11.4	Die Grenzen der Möglichkeiten in Access – Beyond the Limits	1028
 Anhang		1029
A.1	Microsoft SQL Server	1029
A.1.1	Installationsdateien herunterladen	1029
A.1.2	SQL Server Express Edition installieren	1032
A.1.3	Arbeiten nach der Installation	1040
A.1.4	SQL Server Management Studio: Das universelle Werkzeug zur Verwaltung des SQL Servers	1046
A.1.5	Eine Datenbank im SQL Server erstellen	1052
A.1.6	Datenbank trennen und wieder anfügen	1057
A.1.7	Datensicherung einer Datenbank des SQL Servers	1060
A.1.8	Authentifizierung und Berechtigungen	1061
A.2	Access Web Apps in Office 365 oder SharePoint speichern	1069
A.2.1	Access Web Apps benötigen eine SharePoint-Infrastruktur	1070
A.2.2	Beispiel einer Access Web App in SharePoint: Erfassung von Mitarbeitern und Flugbuchungen	1071
A.2.3	Alles SharePoint – und wo ist Access?	1076
Index		1079