

GEOWISSENSCHAFTLICHE DATENBANKEN

(MS-ACCESS 2000)

Skriptum
zur Lehrveranstaltung

von

Michael Rasser

(michael.rasser@paleoweb.net)

Internet-Links zum Thema Datenbanken (auch verfügbar auf <http://paleoweb.net/lv/www-db>)

GEOWISSENSCHAFTLICHE INTERNET - DATENBANKEN IN ÖSTERREICH

Datenbank österreichischer Geowissenschaftler:

<http://www.univie.ac.at/geo>

Directory of Paleontologists of the World:

<http://ipa.geo.ukans.edu:591/Directory/directory.html>

Austrofossil (Verzeichnis der Fossilien in Österreich):

<http://www.oeaw.ac.at/~austrofossil/>

Lithostrat (Lithostrat. Einheiten in Österreich) mit ausführlicher Literaturdatenbank:

<http://www.univie.ac.at/geo/lithstrat>

Literaturdatenbanken der Geologischen Bundesanstalt

(Geolit, Geotext, Geolit, Geokart):

<http://www.geolba.ac.at>

GEOWISSENSCHAFTLICHE INTERNET - DATENBANKEN INTERNATIONAL:

NOW (Neogene of the Old World):

<http://www.helsinki.fi/science/now/>

ETE (Evolution of Terrestrial Ecosystem)

<http://etedata.si.edu/>

ARBEITSGRUPPE DATENBANKEN IN DER PALÄONTOLOGIE:

<http://www.cp-v.de/paldat>

ONLINE KURSBÜCHER

Relationale Datenbanken:

<http://www.ib.hu-berlin.de/~is/rel-db2.htm>

Datenbanken:

<http://www.aedv.cs.tu-berlin.de/edu/se/Datenbanken.html>

Datenbanken mit Access:

<http://www.uni-karlsruhe.de/~za151/access/index.html>

Geoinformatik:

<http://castafiore.uni-muenster.de/vorlesungen/geoinformatik/frames/fsteuer.htm>

Datenbanken I:

<http://www.bib.informatik.th-darmstadt.de/TTT/AI.HTM>

Datenbanksysteme und Aufbau von Datenbanken

<http://193.196.176.31/~riekert/lehre/>

PostScript File zum Herunterladen mit gutem Überblick:

http://www.informatik.uni-muenchen.de/~_braunc/soft.html

Datenbanken:

<http://www.infnet.verein.de/vortraege/rdbms>

THEORIE VON INTERNET-DATENBANKEN

Kleines Web/Datenbanken-Kompendium:

<http://www.simsy.ch/intranet/db.htm>

WWW und Datenbanken:

<http://trumpf-3.rz.uni-mannheim.de/www/sem97s/gutacker>

KOSTENLOSE CGI-SCRIPTS ZUM ERSTELLEN VON WWW-DATENBANKEN

DbMan:

<http://www.gossamer-threads.com/scripts/dbman/>

Testdatenbank: <http://www.gossamer-threads.com/perl/dbman/db.cgi>

"Database Ver 1.0":

<http://websitebuilder.hypermart.net/scripts/details/database.asp>

AdrBook:

<http://www.tesol.net/scripts/AddrBook/>

LINK-SAMMLUNGEN MIT KOSTENLOSEM (INKL. CGI-SCRIPTS):

<http://www.webfree.de/>

<http://www.geldgeier.de/>

<http://linguistic-funland.com/scripts/>

Teil 1: Theorie

Begriffe und Definitionen

Datenbanksystem (DBS): ein System zur Speicherung, Pflege und Auswertung von (meist) umfangreichen Datenmengen, die von mehreren Anwendungen (Programmen) genutzt werden kann. Das DBS besteht aus zwei Komponenten: Datenbank und Datenbank Management System.

Die **Datenbank** stellt die Datenbasis dar. Entspricht der Menge aller gespeicherten Informationen.

Das **Datenbank Management System (DBMS)** ist für die Verwaltung der Daten zuständig.

Um Daten verwalten zu können, muss man sie erst strukturieren. Ein naheliegender Schritt ist es, Daten in Tabellenform darzustellen. Jede Spalte repräsentiert ein Feld mit einem festen Wert, jede Zeile repräsentiert einen Datensatz.

Diese Darstellungsform genügt aber nur für einfachste Anwendungen. Für komplexere Datensammlungen müssen i. A. mehrere Tabellen miteinander verknüpft werden.

Aufgaben eines Datenbanksystems (DBS)

Verwaltung von Daten

Ein DBS muss die Eingabe und Pflege von dauerhaft gespeicherten Daten (Persistenz) ermöglichen. Die Speicherung erfolgt auf der Festplatte oder bei größeren Datenmengen auf Bändern.

Flexible Auswertung von Daten

Ein DBS muss weiters eine flexible Auswertung der Daten vorsehen. Auswertungen können dabei von verschiedener Gestalt sein. Typische Operationen sind hier etwa Suchen und Berechnungen. Viele Aktionen werden mit Hilfe von sog. **Abfragen** ausgeführt.

Unabhängigkeit des Anwendungsprogrammes

Fremdprogramme können auf die in Tabellenform gespeicherten Informationen zugreifen.

Standardisierte Datenbanksprache

Es gibt die standardisierte Datenbanksprache **SQL** (Structured Query Language), die jedes kommerziell verfügbare DBS weitgehend unterstützt. Damit können unabhängige Abfragen formuliert werden.

Performance

Ein wichtiges Ziel ist die Optimierung von Geschwindigkeit und Speicherbedarf. Hierfür gibt es z.B. Indizierungen.

Integritätsprüfung

Das DBS kann schon während der Eingabe Integritätsprüfungen vornehmen. D.h., das System überprüft, ob die eingegebenen Daten vorher definierten Regeln entspricht und vermeidet damit falsche Eingaben (z.B. darf der Preis eines Produktes nicht negativ sein).

Mehrbenutzerbetrieb

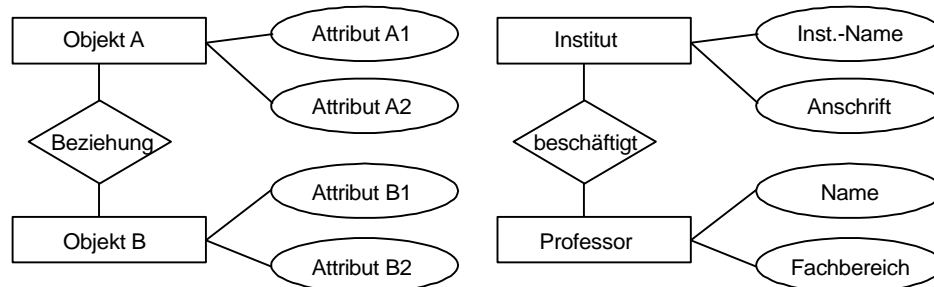
Die meisten kommerziell verfügbaren Systeme unterstützen die Benutzung einer Datenbank durch mehrere Benutzer (z.B. über ein gemeinsames Netzwerk). Das DBS muss dann die Veränderung von Daten durch mehrere Nutzer regeln und so Fehler verhindern.

Zugriffsschutz

Viele Systeme stellen auch Mechanismen zur Verfügung, die den Zugriff auf Daten durch Unbefugte verhindern (siehe z.B. oben). So haben die einen Nutzer nur das Recht, Daten zu lesen, während andere Nutzer auch das Recht haben, Daten zu verändern.

Das Entity-Relationship Modell (Datenmodellierung)

Dieses Modell beschreibt Daten und deren Struktur. Es geht davon aus, dass sich die zu verwaltende Information aus einer Menge von **Objekten (Entities)** beschreiben lässt zwischen denen **Beziehungen (Relationships)** herrschen. Objekte und Beziehungen werden durch **Attribute** beschrieben.

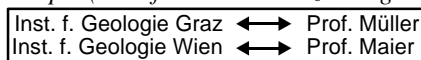


Arten von Beziehungen

1:1 Beziehung

Wenn jedem Institut nur einen Professor beschäftigt ist

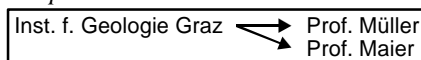
Beispiel (die Pfeile stellen Beziehungen dar):



1:N Beziehung

Wenn an einem Institut (= 1) mehrere Professoren (= N) beschäftigt sind

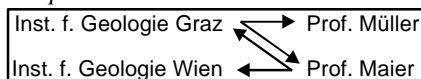
Beispiel:



N:M Beziehung

Wenn es mehrere Institute gibt, an denen Professoren beschäftigt sind, aber diese Professoren auch gleichzeitig an mehreren Instituten beschäftigt sind.

Beispiel:



Relationale Datenbanken

Das vorher besprochene ER-Modell wird in **relationalen Datenbanken** umgesetzt.

Das Grundprinzip von relationalen Datenbanken sind Tabellen, mit denen Objekte und deren Beziehungen modelliert werden können. Tabellen kann man in der Mathematik als sogenannte Relationen auffassen (daher der Name). Das relationale Modell ist dann eine Ansammlung solcher Relationen (Tabellen). (Achtung: diese "Relationen" haben nichts mit den "Relationships" des ER Modelles zu tun!)

	Feldname ←		
	Name	Adresse	Telefon
Feld ←	Huber	Graz	1234
Datensatz ←	Maier	Wien	5678
	Müller	Leoben	9101

Heute werden praktisch nur noch relationale Datenbanken verwendet.

Zu den häufigsten Anwendungen zählen **Access, DBase, Oracle, FoxPro**. Aber auch eine in **Excel** erstellte Tabelle stellt eine Datenbank dar!!!!!!!

Ein paar Worte to ArcView - Arc/Info - GIS

(Quelle: [http:// castafiore.uni-muenster.de/vorlesungen/av_tutor_1.2](http://castafiore.uni-muenster.de/vorlesungen/av_tutor_1.2))

Ein **Geo-Informationssystem (GIS)** ist ein rechnergestütztes System, das aus Hardware, Software, Daten und Anwendern besteht. Mit ihm können **raumbezogene** Daten digital erfasst und redigiert, gespeichert und reorganisiert, modelliert und analysiert sowie alphanumerisch und graphisch präsentiert werden.

ARC/INFO ist ein Software-Paket für ein geographisches Informationssystem (GIS) von Environmental Systems Research Institute (ESRI), dem Hersteller von **ArcView**. ARC/INFO wird weltweit von Tausenden verschiedenen Firmen und Organisationen dazu verwendet, geographische Informationen zu handhaben, zu verwalten und zu analysieren.

Wesentlich ist die Möglichkeit, raumbezogene Daten zu verwalten, wozu auch verschiedene Arten von Grafiken gehören. Attribute, die geographische Objekte beschreiben, werden in tabellarischen Dateien oder externen Datenbanksystemen abgespeichert. ArcView-Benutzer können auf praktisch alle im ARC/INFO-Format abgespeicherten raumbezogenen Daten zum Zwecke der Desktop-Abbildung und Analyse zugreifen.

Aufgaben eines Geo-Informationssystems

⇒ Erfassung raumbezogener Daten

Unter der Eingabe versteht man alle Techniken zur Erfassung von geometrischen Daten, z.B. durch Digitalisierung, Scannen, Fernerkundung.

⇒ Verwaltung raumbezogener Daten

Als Voraussetzung für die Manipulation und die digitale Auswertung sind die erfassten Daten in einem GIS in digitaler Form in Datenbanken zu verwalten.

⇒ Analyse raumbezogener Daten

Im Mittelpunkt der raumbezogenen Datenverarbeitung mit GIS steht die vielfältige Datenanalyse. Die Analyse dient primär der Gewinnung neuer Informationen bzw. soll die vorhandenen Information nach bestimmten Kriterien aufschlüsseln. Die Algorithmen unterscheiden sich hinsichtlich der Geometrie auf der sie arbeiten. Für Vektordaten müssen andere Analysealgorithmen angewandt werden als für Rasterdaten. Außerdem müssen die Sachdaten als nicht-geometriebezogene Daten meist mit Funktionen relationaler Datenbank-Managementsysteme (SQL) ausgewertet werden.

⇒ Präsentation raumbezogener Daten

Zur Visualisierung von Geo-Informationen bieten sich eine Reihe von Darstellungsformen an. Im Mittelpunkt steht die Karte, die am Bildschirm oder auf einem analogen Druckträger wiedergegeben wird. Sie bildet ein zweidimensionales, graphisches, maßstabsgebundenes Modell, das die Realität verkleinert und generalisiert abbildet.

Andere Präsentationsformen sind kartenverwandte Darstellungen wie Profile, Schaubilder oder Ansichten sowie Bildsequenzen und Animationen. Darüberhinaus können Auswertergebnisse in Tabellen- oder Textform sowie als Diagramme wiedergegeben werden. Hinzu kommt die Möglichkeit Objekte und Situationen durch Fotografien zu dokumentieren.

Ein Bericht, der über einen Gesamtzusammenhang informieren will, wird in der Regel verschiedene Präsentationsmittel nutzen. Um eine komplexe Situation mit ihren Raumbezügen zu erläutern, kann auch ein Poster geeignet sein.

Teil 2: Access 2000

Einfache Datenbanken mit Assistenten erstellen

Beispiel 1

Aufgabe: Erstellung einer einfachen Adressen-Datenbank mit Hilfe der Datenbank-Assistenten

1. Tabelle erstellen

Nach dem Öffnen von Access 2000 *Leere Datenbank* auswählen. Dateiname: **Adressen**.

- Es erscheint nun das Hauptmenü mit den sog. '**Objekten**':

(Achtung: Diese 'Objekte' haben nichts mit den Objekten im Sinne des Entity-Relationship Modelles zu tun!)

Tabellen	Tabellen stellen die Struktur der einzelnen Datensätze dar. Jeder Datensatz ist in einer Tabellenzeile gespeichert.
Abfragen	Mit Hilfe von Abfragen werden Daten aus den Tabellen herausgefiltert. Damit kann gezielt auf Teile des Datenbestandes zugegriffen werden.
Formulare	Mit Formularen können Karteikarten gedruckt und Bildschirmmasken erstellt werden, die zur Anzeige von Daten aus Tabellen und Abfragen dienen.
Berichte	Ein Bericht ist ein speziell für den Ausdruck von Listen entwickeltes Formular.
Makros	Makros fassen Folgen von nacheinander ausgeführten Befehlen unter einem Namen zusammen, sodass Arbeitsgänge abgekürzt werden können.
Module	Module sind Prozeduren, die in der Programmiersprache von Access erstellt wurden.

Dann in 'Objekte' *Tabellen* markieren und *Neu*. Nun wählt man den *Tabellen-Assistenten*.

Im folgenden Fenster erscheinen vorgegebene und vordefinierte Beispieltabellen (links) und entsprechende Felder (rechts).

Nun wählt man die entspr. benötigten Felder, wie Vorname, Nachname, Adresse, Bundesland, etc. aus, indem man das jeweilige Feld markiert und auf das ">" Symbol klickt.

Dann klickt man auf *weiter*, gibt einen Tabellennamen ein (dieser ist unabhängig vom Dateinamen!!) und wählt zuletzt *Direkt Daten in die Tabelle eingeben* an.

Nun erhalte ich eine simple Tabelle, in die ich meine Daten eingeben kann. Sobald ich Daten eingabe, erscheint darunter eine neue leere Zeile für weitere Daten.



Übung: Geben Sie nun ein paar beliebige Datensätze ein!

Das Ergebnis sieht, z.B. so aus:

AdressenlisteID	Vorname	Nachname	Adresse	Bundesland
1	Anton	Huber	Wien	Wien
2	Ewald	Steuber	Graz	Steiermark
3	Bernhard	Maier	Wien	Wien
4	Erwin	Mayer	Klagenfurt	Kärnten
5	Hans	Rauscher	Graz	Steiermark
*	(AutoWert)			

Datensatz: 5 von 5

Übung: Einfache Abfrage in der Tabelle durch Filter

Klicke auf das Symbol  - klicke dann mit der Maus in ein Feld (z.B. Bundesland) und wähle dann einen vorhandenen Wert aus. Nach dem klicken auf das Symbol  werden nur noch jene Datensätze angezeigt, die jenem Wert entsprechen. Nach nochmaligem klicken werden wieder alle Datensätze angezeigt.

2. Formular erstellen

Übung: Erstellung einer Eingabemaske mit dem Assistenten

Schließe die Tabelle und wähle im Hauptfenster *Formulare - Neu - Formular-Assistent*.

Wähle alle Felder aus (ausgenommen das ID-Feld), wähle dann *einspaltig* und ein entsprechendes Format.

Schließlich einen Namen für das Formular auswählen (hat wieder nichts mit Datein-Namen zu tun) und *Fertigstellen*.

Schließlich erhalte ich das fertige Formular mit einem Datensatz-Navigator in der Fuß-Zeile.



3. Abfrage erstellen

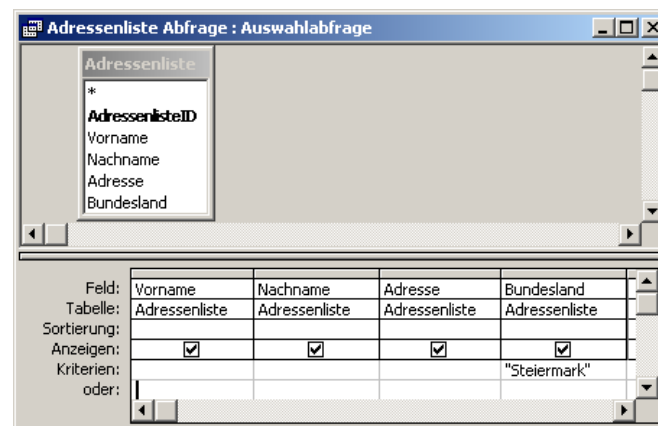
Wähle *Abfrage - Neu - Auswahlabfrage-Assistent*.

Wähle die entsprechenden Felder aus und klicke schließlich *Fertigstellen*.


Wir erhalten schließlich eine Tabelle, die aussieht wie unsere Originaltabelle.

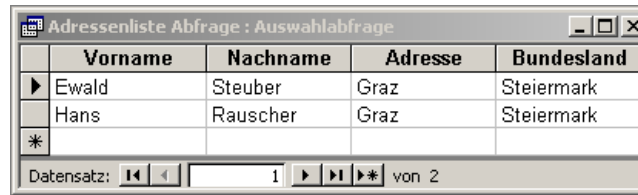
Gehe nun über das Symbol  in die **Entwurfsansicht**.

Hier können die Abfrageoptionen ausgewählt werden, z.B.:



Feld:	Vorname	Nachname	Adresse	Bundesland
Tabelle:	Adressenliste	Adressenliste	Adressenliste	Adressenliste
Sortierung:				
Anzeigen:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kriterien:				"Steiermark"
oder:				

Gehe nun durch klicken auf das Symbol  in die Datenblattansicht und das Ergebnis sieht so aus:



	Vorname	Nachname	Adresse	Bundesland
▶	Ewald	Steuber	Graz	Steiermark
■	Hans	Rauscher	Graz	Steiermark
*				

Datensatz: 1 von 2

4. Berichte erstellen

Wähle *Berichte - Neu - Berichtsassistent* und wähle die entsprechenden Felder aus.

Bei der Frage nach einer *Gruppierungsebene* wählen wir nichts aus. Wir wählen dann nach Geschmack beliebige Layout-Optionen aus und sagen schließlich *Fertigstellen*.

Das Ergebnis ist nun eine für den Ausdruck geeignete Liste.

Erstellung komplexer Datenbanken

Beispiel 2

Aufgabe: Datenbank mit geologischen Aufschlüssen (Lokalitäten) und den dazugehörigen Proben erstellen.

Lokalität Fuchsofen am Krappfeld (Kärnten). Alter: Unter-Eozän

Proben:

kp01-1 (mikritischer Kalk)

kp01-2 (Kalksandstein)

kp01-3 (Kalk)

Lokalität Unterstein am Krappfeld (Kärnten). Alter: Mittel-Eozän

Proben:

unt01-1 (Kalksandstein) ANALYSE VORHANDEN

unt01-2 (mikritischer Kalk)

Lokalität Eggenburg (Niederösterreich). Alter: Unter-Miozän

Proben:

egg01-1 (Kalk) ANALYSE VORHANDEN

egg01-2 (Kalksandstein)

egg01-3 (mikritischer Kalk)

Übung: Überlege folgende Punkte

- Welche Felder werden benötigt?
- Plane die Struktur der Datenbank
- Überlege zuerst, wie diese Datenbank in einer einzigen Tabelle aussehen könnte.
- Dann überlege dir die Datenbankstruktur im Sinne des ER-Modells und zeichne ein ER - Diagramm, wie in der Theorie besprochen. Was sind die Objekte, was die Beziehungen und was die Attribute?
- Mit welcher Art von Beziehungen haben wir es zu tun (1:1, 1:N, N:M)?

Ansatz: Zur Vermeidung der oben diskutierten Datenredundanzen werden zwei Tabellen angefertigt und miteinander verknüpft.

Wichtig: Jede Probennummer ist einzigartig, wiederholt sich nicht und ist eindeutig mit einer Lokalität verknüpft.

5. Neue Datenbank erstellen

- MS Access 2000 öffnen und 'leere Access-Datenbank' auswählen.
- Aussagekräftigen Datei-Namen angeben.

in unserem Beispiel: "**Proben**"

Als erstes wird eine neue Tabelle erstellt.

6. Tabellen anlegen

6.1. Erstellen einer Tabelle

Wir beginnen mit der Erstellung der Tabelle mit den Lokalitäten

Im Hauptmenü wählt man unter *Objekte* die Option *Tabellen* aus und dann *Neu*.

Im folgenden Fenster *Neue Tabelle* erscheinen fünf Optionen. *Datenblattansicht* und *Entwurfsansicht* legen eine neue leere Tabelle an. Mit *Tabellenassistent* können vordefinierte Tabellen (z.B. Adressenkarteien, Kundenkarteien) angelegt werden. Mit *Tabelle importieren* können vorhandene Tabellen (z.B. aus Excel) importiert werden. Mit *Tabelle verknüpfen* können Tabellen aus anderen Datenbanken verknüpft werden (dazu später).

Für unser Beispiel wählen wir *Datenblattansicht*.

6.2. Tabellen in der Datenblattansicht

In der Datenblattansicht sehen wir eine neue leere Tabelle, wie es sie auch in Excel gibt. Die einzelnen Zeilen entsprechen (noch nicht vorhandenen) Datensätzen, die einzelnen Spalten repräsentieren die Felder. In der Kopfzeile sieht man die Namen der Felder (zur Zeit Feld1, Feld2, etc.).

Jede Tabelle kann in der Datenblattansicht und der Entwurfsansicht dargestellt werden. Zum wechseln klickt man in der Menüzeile auf das Ansicht-Symbol in der linken oberen Ecke.

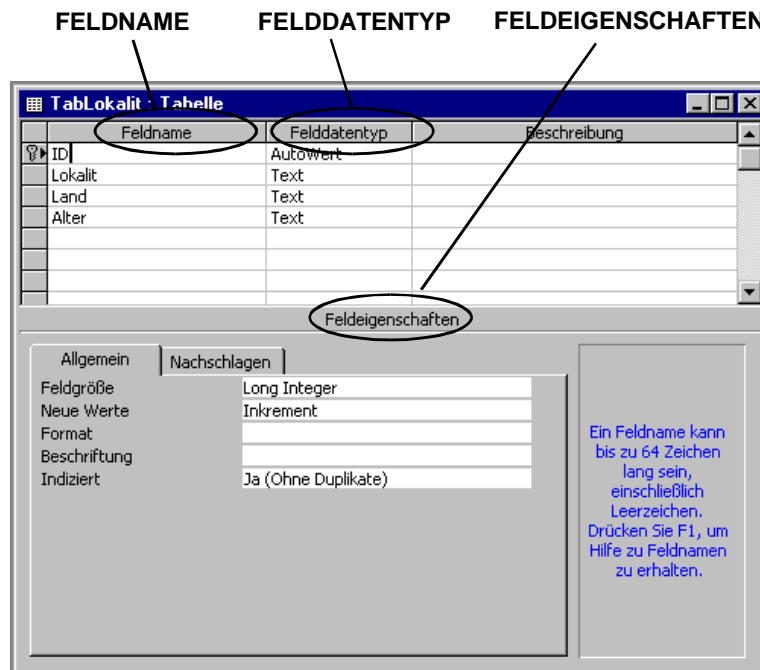
Datenblattansicht:  , Entwurfsansicht: 

6.3. Tabellen in Entwurfsansicht definieren

In der *Entwurfsansicht* müssen nun die einzelnen Felder der Tabelle definiert werden.

Der wichtigste Bestandteil einer Tabelle ist der **Primärschlüssel**, der jeden einzelnen Datensatz unzweifelhaft definiert und in der Tabelle nur einmal vorkommen darf. Wird z.B. in einer Adressendatei jeder Datensatz durch einen Nachnamen definiert, ergibt sich das Problem, dass ein Name mehrfach vorkommen oder sich (z.B. durch Heirat) verändern kann. Daher wird man als Primärschlüssel eine eindeutiges Feld wie Kundennummer auswählen, oder man fügt ein eigenes Feld mit dem *Feldnamen ID* ein. Im *Felddatentyp* wählt man dann *AutoWert*. Das bedeutet, dass beim Einfügen eines neuen Datensatzes automatisch eine neue laufende Zahl eingefügt wird, die den *Primärschlüssel* darstellt und in der Tabelle jeweils nur einmal vorkommt. Der Vorteil ist, dass nun Tabellen nur über diese ID verknüpft werden und der Inhalt des jeweiligen Datensatzes beliebig korrigiert werden kann, ohne dass Probleme bei der Verknüpfung auftreten.

Ein weiterer wichtiger Punkt ist die **Indizierung**. Felder, nach denen eine Datenbank häufig durchsucht werden (im Falle einer Adressendatei z.B. der Nachname), sollten indiziert werden, weil das Suchen in indizierten Feldern schneller abläuft. Der Index ist mit dem Stichwortverzeichnis eines Buches vergleichbar. Für indizierte Felder wird eine spezielle Tabelle erstellt, die mit Hilfe des Indexbegriffes das schnelle Auffinden eines Datensatzes erlaubt. Insbesondere sollte ein Feld indiziert werden, über dessen Inhalt zwei Tabellen miteinander verknüpft werden.



Feldname

Zunächst wird jedem Feld ein eindeutiger *Feldname* zugewiesen.

Für unser Beispiel benötigen wir in der Lokalitäten-Tabelle (1) Name der Lokalität, (2) Bundesland, (3) Alter. Die Feldnamen sollten eindeutig und prägnant sein; Vorschlag: **Lokalität**, **Land** und **Alter**.

Am häufigsten wird man nach **Lokalität** suchen. Es empfiehlt sich daher, diese zu indizieren.

Nachdem der Name der Lokalität immer eindeutig sein sollte, könnte man diesen als *Primärschlüssel* verwenden. In unserem Beispiel werden wir aber ein eigenes Primärschlüssel-Feld einfügen:

Wenn man ein eigenes *Primärschlüsselfeld* (*ID*) einfügen will, benennt man das erste (= oberste) Feld "ID", klickt dann mit der rechten Maustaste darauf und wählt *Primärschlüssel*. Neben dem Feld erscheint dann ein Schlüsselssymbol. Außerdem muss als *Felddatentyp* der Wert *AutoWert* angegeben werden, damit jeder neue Datensatz automatisch eine neue Nummer bekommt. Bei *Indiziert* muss *Ja (Ohne Duplikate)* stehen (geschieht automatisch), damit jede ID einzigartig ist.

In unserem Fall werden wir das *Primärschlüsselfeld* als **Lok-ID** bezeichnen um spätere Verwechslungen zu vermeiden.

Nun vergibt man die weiteren Feldnamen. Will man ein *Feld einfügen oder löschen*, klickt man mit der rechten Maustaste in ein Feld und wählt dann die entspr. Option aus.

Zu jedem Feldnamen kann man nun den Felddatentyp und die Feldeigenschaften definieren. Dazu klickt man auf den entspr. Feldnamen und wählt dann den Felddatentyp (rechts davon) und die Feldeigenschaften (unten) aus.

Felddatentyp

Text	Zeichenketten von maximal 255 Zeichen
Memo	Texte mit maximal 64000 Bytes Länge, eignet sich für Bemerkungs-Feld. <i>Achtung: dieses Feld kann nicht indiziert werden.</i>
Zahl	Numerische Werte (Ganz- oder Dezimalzahlen), 1, 2, 4 oder 8 Bytes
Datum/Zeit	Datum und Uhrzeit, 8 Bytes
Währung	Geldbeträge, 8 Bytes
AutoWert	Ein Datensatzzähler, den Access jedem Datensatz zuordnet. <i>Kann für das ID-Feld als Primärschlüssel verwendet werden.</i>
Ja/Nein	Auswahl zwischen zwei Werten. Z.B. vorhanden/nicht vorhanden, oder männlich/weiblich.
OLE-Objekt	Z.B. Grafiken

Feldeigenschaften

Feldgröße	Gibt beim Datentyp " <i>Text</i> " die maximale Anzahl Eingabezeichen an, bei " <i>Zahl</i> " den Zahlentyp (Byte, Integer usw.). Die vorgegebene Größe entscheidet über die Größe der gesamten Datenbank, da für jeden Datensatz in dem entspr. Feld die vorgegebene Größe vorgesehen wird. Zahlentypen: <i>Byte</i> = ganze Zahl zw. 0 und 255 (Größe 1 Byte); <i>Integer</i> = ganze Zahl zw. -32.768 und 32.767 (2 Byte); <i>Long Integer</i> = ganze Zahl zw. -2 mrd. und 2 mrd. (4 Byte); <i>Single</i> = Dezimalzahlen mit 6stelliger Genauigkeit (4 Byte); <i>Double</i> = Zahlen, auch Dezimalzahlen mit 10stelliger Genauigkeit (8 Byte); <i>Replikations-ID</i> = von Access vergebene Identifikationsnummer (16 Byte); <i>Dezimal</i> = Dezimalzahlen mit wählbarer Genauigkeit (variabel).
Format	Gibt bei Zahlen und Datumsangaben ein Anzeigeformat (z.B. Währung, Prozent usw.) an
Dezimalstellen	Gibt bei Zahlen- oder Währungsangaben die Anzahl der anzuzeigenden Dezimalstellen an
Eingabeformat	Zeigt Formatierungszeichen bei Währungs- und Datumsangaben im Eingabefeld an
Beschriftung	Legt statt des Feldnamens eine andere Standardbeschriftung für dieses Feld fest. Wird nichts eingegeben, so wird der oben definierte Feldname verwendet (z.B. in Formularen)
Standardwert	Trägt in neue erzeugte Datensätze automatisch einen Vorschlag ein, der dann aber noch verändert werden kann
Gültigkeitsregel	Legt Bedingungen fest, unter denen Access Gültigkeitsmeldung eingegebene Werte akzeptieren oder ablehnen soll
Gültigkeitsmeldung	Meldung, die am Bildschirm erscheint, wenn die Gültigkeitsregel verletzt wird.
Eingabe Erforderlich	Legt fest, ob das betreffende Feld vom Benutzer leer gelassen werden darf. Wird hier "ja" ausgewählt, so wird man bei der Eingabe darauf hingewiesen, wenn man vergessen hat, dieses Feld auszufüllen
Leere Zeichenfolge	Wird <i>Eingabe Erforderlich</i> mit "ja" angegeben, kann keine Datensatz angelegt werden, ohne dass das Feld ausgefüllt wird. Wenn <i>Leere Zeichenfolge</i> mit "ja" angegeben ist, kann ich das Feld trotzdem leer lassen, muss aber in das Feld gehen und dort die Leertaste drücken.
Indiziert	Ist ein Feld indiziert, dann laufen Suchvorgänge in diesem Feld schneller ab

Für unser Beispiel wird man für alle Felder *Eingabe erforderlich* = ja auswählen. Nur bei *Alter* wird man auch *Leere Zeichenfolge* = ja eintragen, falls das Alter einer Lokalität nicht bekannt ist.

6.4. Abspeichern der Tabelle

Zum Abspeichern der Tabelle im Menü *Datei - Speichern unter* aus und vergibt einen aussagkräftigen Namen für die Tabelle.

Für unser Beispiel empfiehlt sich z.B. "*TabLokalit*". Durch den Zusatz "Tab" wird der Überblick bei zukünftigen Operation übersichtlicher, da man sofort erkennt, dass es sich um eine Tabelle handelt.

Diese Tabelle wird nicht als eigene Datei gespeichert, sondern innerhalb der beim Beginn ausgewählten Datenbank-Datei (mit der Endung *.mdb).

Wird diese Tabelle nun geschlossen, scheint sie im Hauptmenü unter den 'Objekten' als Tabelle auf.

(Anmerkung: Wenn man die Tabelle für andere Anwendungen benötigt, kann man sie auch für sich als eigene Datei (z.B. Excel) exportieren. Dazu wählt man *Datei - Exportieren*.)

6.5. Anfertigen einer neuen Tabelle

Wir schließen nun die vorhin angefertigte Tabelle und fertigen nach dem selben Prinzip eine zweite Tabelle an.

Nachdem wir nun die Tabelle mit den Lokalitäten haben, brauchen wir noch eine zweite Tabelle mit den Probennummern und ihrer Charakterisierung.

Folgende Angaben benötigen wir: (1) Proben-Nummer, (2) Gesteinscharakterisierung; außerdem kann man noch (3) Probe bereits analysiert: ja/nein einfügen.

Folgende Feldnamen ergeben sich daraus: **Probe**, **Gestein**, **Analyse**. Außerdem empfiehlt es sich immer, ein Feld **Bemerkung** einzufügen.

Nachdem jede Probennummer nur einmal vorkommt, könne wir **Probe** als **Primärschlüssel** definieren.

Außerdem muss natürlich *Eingabe Erforderlich - Ja* angegeben werden. **Achtung: nicht auf AutoWert stellen!!**

Das Feld **Analyse** beinhaltet nur die Angaben "Ja" oder "Nein". Als Felddatentyp wird daher *Ja/Nein* ausgewählt.

Bemerkung soll größere Mengen Text aufnehmen können und wird daher als *Memo* definiert!

Wenn man ein vorhandenes Feld (bei uns *Probe*) als **Primärschlüssel** definieren will, klickt man mit der rechten Maustaste darauf und wählt *Primärschlüssel*. Neben dem Feld erscheint dann ein Schlüsselymbol.

In diesem Fall muss für den Feldnamen *Probe* in den Feldeigenschaften bei Indiziert: Ja (Ohne Duplikate) angegeben werden. Dies verhindert, dass eine Probennummer zweimal vergeben werden kann.

Abspeichern der Tabelle unter dem Namen "*TabProben*".

7. 1:N Beziehungen

7.1. Verknüpfung des Primärschlüssels

Unsere Proben-Datenbank ist ein typisches Beispiel für eine 1:N Beziehung. Jeder Lokalität ("1") sind mehrere Proben-Nummern ("N") zugeordnet.

Eine 1:N Beziehung wird hergestellt, indem das *Primärschlüsselfeld* der Tabelle "1" in die Tabelle "N" als *neues Feld* eingefügt wird.

Das Primärschlüsselfeld wird verwendet, weil es sich um einen eindeutigen Schlüssel handeln muss.

Wir öffnen die Tabellen *TabLokalit* und *TabProben* in der *Entwurfsansicht* (z.B. Tabelle markieren und auf *Entwurf* klicken).

Nun wird in *TabLokalit* (= Tabelle "1") die gesamte Zeile des Primärschlüsselfeldes *Lok-ID* markiert und kopiert (Menü Bearbeiten - Kopieren). Dann wird dieses in *TabProben* (= Tabelle "N") in der letzten Zeile eingefügt (Menü Bearbeiten - Einfügen).

Wichtig: Der Felddatentyp des eingefügten Feldes muss **von *AutoWert* auf *Zahl*** umgestellt werden (weil dieser Wert ja nun aus der anderen Tabelle kommt). Außerdem sollte *Indiziert* auf *Nein* oder *Ja (Duplikate möglich)* gestellt werden.

Für uns bedeutet dies: **in *TabProben*** wird nun der Felddatentyp von ***Lok-ID* auf *Zahl*** umgestellt sowie die ***Feldgröße* auf *LongInteger***. Dies ist nötig, weil diese Nummer ja aus *TabLokalit* kommt und keinesfalls in *TabProben* neu erstellt werden soll! Außerdem sollte *Indiziert* auf *Nein* oder *Ja (Duplikate möglich)* gestellt werden.

7.2. Beziehungen definieren

Nachdem nun der Primärschlüssel aus Tabelle "1" in die Tabelle "N" eingefügt wurde, muss noch die Beziehung zwischen den Tabellen definiert werden.

Dazu wählt man im Menü *Extras - Beziehungen* und erhält ein Fenster mit den vorhandenen Tabellen. Man markiert nun die entspr. Tabellen und klickt auf *Hinzufügen*. Hat man eine Tabelle zuviel hinzugefügt, markiert man diese einfach und drückt auf der Tastatur *Entfernen*.

Nun klickt man mit der Maus in Tabelle "1" auf den *Primärschlüssel*, und ziehen ihn auf das selbe Feld, das Sie in Tabelle "N" eingefügt haben, bevor Sie die Maustaste wieder loslassen.

In unserem Beispiel klickt man in *TabLokalit* auf *Lok-ID* und zieht es in *TabProben* auf *Lok-ID*.

Nun erscheint das Fenster *Beziehungen bearbeiten* und zeigt die verknüpften Tabellen und Felder an:



Optionen:

Referenzielle Integrität ist ein Regelsystem, mit dessen Hilfe Access sicherstellt, dass Beziehungen zwischen Datensätzen in Detailtabellen gültig sind und dass verknüpfte Daten nicht versehentlich gelöscht oder geändert werden.

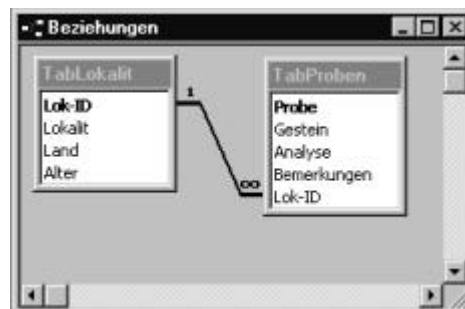
Wenn **Aktualisierungsweitergabe an Detailfeld** *zusätzlich* aktiviert ist, aktualisiert Access bei jeder Änderung des Primärschlüssels eines Datensatzes der Mastertabelle (= Tab. "1", bei uns *TabLokalit*) automatisch den Primärschlüssel in allen Detaildatensätzen (= Tab. "N", bei uns *TabProben*) auf den neuen Wert.

Wenn **Löschweitergabe an Detaildatensatz** *zusätzlich* aktiviert ist, löscht Access bei jedem Löschen von Datensätzen in der Mastertabelle (Tab. "1", bei uns *TabLokalit*) automatisch die Detaildatensätze in der Detailtabelle (Tab. "N", bei uns *TabProben*). Dadurch werden Karteileichen verhindert.

Dies sind Beispiele für die in der Theorie besprochene Integritätsprüfung.

ACHTUNG: Wenn Sie Datensätze löschen, kann dieser Löschvorgang nicht mehr rückgängig gemacht werden. Daher ist es wichtig zu bedenken, dass bei der Aktivierung von 'Löschweitergabe an Detaildatensatz' folgende Konsequenz hat: Wenn ich eine Lokalität lösche, werden sämtliche mit ihr verknüpften Proben ebenfalls gelöscht! Zuerst muß daher überlegt werden, ob ich das überhaupt will.

Wenn Sie nun *Erstellen* klicken, erhalten Sie die unten gezeigte Verknüpfung, wobei das Unendlich-Zeichen (Brille) für das "N" in der 1:N Beziehung steht:



8. Dateneingabe über die Tabelle

Nach erfolgter Verknüpfung können die Daten eingegeben werden. Dazu öffnet man die Haupttabelle (Tabelle "1") und gibt hier die Daten ein. Zu jedem Datensatz erscheint links ein "+". Wird dieses Plus angeklickt, öffnet sich ein Unterdatenblatt. Dieses Unterdatenblatt entspricht Teilen der Tabelle "N", in welchem nun ebenfalls die entsprechenden Daten eingegeben werden können.

Um von einem Feld ins Nächste zu springen, können Sie die Tabulator- oder Eingabe-Taste verwenden.

Einen neuen Datensatz fügen Sie ein, indem Sie in die letzte Zeile mit dem Stern-Symbol klicken und dort Ihre Daten eingeben. Dann wird automatisch darunter eine neue leere Zeile für weitere Daten eingegeben.

Am untersten Fenster sehen Sie folgende Leiste, mit der Sie zwischen den Datensätzen navigieren können:



Übung: Öffnen Sie die Tabelle *TabLokalit* und geben Sie die Daten aus dem Beispiel ein

So sehen *TabLokalit* und *TabProben* nach der Dateneingabe aus. **Beachte die Übereinstimmungen in den Feldern Lok-ID**

Lok-ID	Lokalit	Land	Alter																				
3	Eggenburg	Niederösterreich	Unter-Miozän																				
<table border="1"> <thead> <tr> <th>Probe</th> <th>Gestein</th> <th>Analyse</th> <th>Bemerkung</th> </tr> </thead> <tbody> <tr> <td>egg01-1</td> <td> Kalk</td> <td><input checked="" type="checkbox"/></td> <td></td> </tr> <tr> <td>egg01-3</td> <td> mikritischer Kalk</td> <td><input type="checkbox"/></td> <td></td> </tr> <tr> <td>*</td> <td></td> <td><input type="checkbox"/></td> <td></td> </tr> </tbody> </table>				Probe	Gestein	Analyse	Bemerkung	egg01-1	Kalk	<input checked="" type="checkbox"/>		egg01-3	mikritischer Kalk	<input type="checkbox"/>		*		<input type="checkbox"/>					
Probe	Gestein	Analyse	Bemerkung																				
egg01-1	Kalk	<input checked="" type="checkbox"/>																					
egg01-3	mikritischer Kalk	<input type="checkbox"/>																					
*		<input type="checkbox"/>																					
1	Fuchsofen	Kärnten	Unter-Eozän																				
<table border="1"> <thead> <tr> <th>Probe</th> <th>Gestein</th> <th>Analyse</th> <th>Bemerkung</th> </tr> </thead> <tbody> <tr> <td>kp01-1</td> <td> Mikritischer Kalk</td> <td><input type="checkbox"/></td> <td></td> </tr> <tr> <td>kp01-2</td> <td> Kalksandstein</td> <td><input type="checkbox"/></td> <td></td> </tr> <tr> <td>kp01-3</td> <td> Kalk</td> <td><input type="checkbox"/></td> <td></td> </tr> <tr> <td>*</td> <td></td> <td><input type="checkbox"/></td> <td></td> </tr> </tbody> </table>				Probe	Gestein	Analyse	Bemerkung	kp01-1	Mikritischer Kalk	<input type="checkbox"/>		kp01-2	Kalksandstein	<input type="checkbox"/>		kp01-3	Kalk	<input type="checkbox"/>		*		<input type="checkbox"/>	
Probe	Gestein	Analyse	Bemerkung																				
kp01-1	Mikritischer Kalk	<input type="checkbox"/>																					
kp01-2	Kalksandstein	<input type="checkbox"/>																					
kp01-3	Kalk	<input type="checkbox"/>																					
*		<input type="checkbox"/>																					
2	Unterstein	Kärnten	Mittel-Eozän																				
<table border="1"> <thead> <tr> <th>Probe</th> <th>Gestein</th> <th>Analyse</th> <th>Bemerkung</th> </tr> </thead> <tbody> <tr> <td>unt01-1</td> <td> Kalksandstein</td> <td><input checked="" type="checkbox"/></td> <td></td> </tr> <tr> <td>unt01-2</td> <td> mikritischer Kalk</td> <td><input type="checkbox"/></td> <td></td> </tr> <tr> <td>*</td> <td></td> <td><input type="checkbox"/></td> <td></td> </tr> </tbody> </table>				Probe	Gestein	Analyse	Bemerkung	unt01-1	Kalksandstein	<input checked="" type="checkbox"/>		unt01-2	mikritischer Kalk	<input type="checkbox"/>		*		<input type="checkbox"/>					
Probe	Gestein	Analyse	Bemerkung																				
unt01-1	Kalksandstein	<input checked="" type="checkbox"/>																					
unt01-2	mikritischer Kalk	<input type="checkbox"/>																					
*		<input type="checkbox"/>																					
* (AutoWert)																							

Probe	Gestein	Analyse	Bemerkung	Lok-ID
egg01-1	Kalk	<input checked="" type="checkbox"/>		3
egg01-2	Kalksand	<input type="checkbox"/>		
egg01-3	mikritischer Kalk	<input type="checkbox"/>		3
kp01-1	Mikritischer Kalk	<input type="checkbox"/>		1
kp01-2	Kalksandstein	<input type="checkbox"/>		1
kp01-3	Kalk	<input type="checkbox"/>		1
unt01-1	Kalksandstein	<input checked="" type="checkbox"/>		2
unt01-2	mikritischer Kalk	<input type="checkbox"/>		2
		<input type="checkbox"/>		

Sie hätten aber auch zuerst in der *TabLokalit* alle Lokalitäten eingeben können, dann die *TabProben* öffnen und dort sämtliche Proben eingeben können. Dann geben Sie in *TabProben* im Feld *Lok-ID* den entsprechenden Primärschlüssel (ebenfalls *Lok-ID*) der zugeordneten Lokalität aus *TabLokalit* ein. Damit haben Sie ebenfalls eine Verknüpfung erstellt und dasselbe Ergebnis erreicht, allerdings ist diese Methode umständlicher und birgt Fehlerquellen in sich.

9. Tabellen - Layout, Sortieren, Filtern

Ähnlich wie in Excel können über das Menü *Format Text*, *Spalten*, etc. verändert und formatiert werden.

Mit folgenden Symbolen können Sie die Datensätze sortieren und filtern, oder Sie wählen im Menü *Datensätze* die entspr. Option aus



Übung: Verändern Sie das Layout der Tabelle (Hintergrundfarbe, Zeichen, etc.) und probieren Sie die Filter-Vorgänge aus.

10. Eine Abfrage erstellen

Abfragen ("Queries") basieren auf bereits vorhandenen Tabellen (oder auch anderen Abfragen) einer Datenbank.

Zum Erstellen einer neuen Abfrage wählen Sie im Hauptmenü aus den 'Objekten' *Abfrage* und klicken auf *Neu*. Sie erhalten ein Fenster mit folgenden Optionen:

Entwurfsansicht öffnet eine leere Auswahlabfrage; bei dieser wird aus einem gesamten Datenbestand eine gezielte Auswahl getroffen. Das Ergebnis lässt sich am Bildschirm anzeigen oder ausdrucken.

Die **Assistenten** vereinfachen die Erstellung einer Abfrage.

Kreuztabellenabfrage ermöglicht die Darstellung der Daten ähnlich einer Kalkulationstabelle. So können große Datenmengen in übersichtlicher Form angezeigt werden.

Duplikatsuche prüft, ob Duplikate in gewissen Feldwerten vorhanden sind.

Inkonsistenzsuche sucht nach Datensätzen in einer Tabelle, die mit keinen Datensätzen in anderen Tabellen im Zusammenhang stehen (Karteileichen).

Für unser Beispiel schließen wir alle Tabellen und wählen eine *Neu - Entwurfsansicht*. In dem folgenden Fenster kann man auswählen in welchen Tabellen gesucht werden soll.

Wir wählen *TabLokalit* und *TabProben* aus und erhalten zwei kleine Fenster mit der Verknüpfung der beiden Tabellen, die wir über *Beziehung definieren* (s.o.) verknüpft haben.

Aufgabe: Ich möchte wissen, für welche Proben in Beispiel 2 ich bereits Analysen durchgeführt habe (in unserem Beispiel waren das egg01-1 und unt01-1).

Ansatz: Wählen Sie aus den im oberen Fenster angezeigten Tabellen das Feld *Lokalit* aus und ziehen es in die erste Spalte der unteren Tabelle in die Zeile 'Feld'. In die nächste Spalte kommt *Probe*, dann *Gestein* und *Analyse*

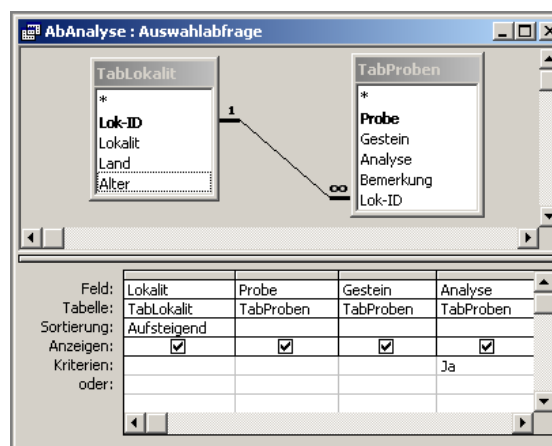
Über *Sortierung* kann ich die Sortierungsart wählen und das Feld, nach dem sortiert wird.

Bei *Anzeigen* kann ich auswählen, ob das eingefügte Feld im Ergebnis auch angezeigt wird (d.h., ich kann z.B. nach einem Begriff in einem bestimmten Feld suchen, möchte das Feld aber nicht angezeigt haben).

Bei *Kriterien* geben Sie nun bei *Analyse* "Ja" ein (wir erinnern uns: *Analyse* ist ein Ja/Nein Feld).

Speichern Sie diese Abfrage als *AbAnalyse* ab.

Unsere **Abfrage** in der **Entwurfsansicht** sieht nun so aus:



Das **Ergebnis** in der **Tabellenansicht** sieht so aus:

Lokalit	Probe	Gestein	Analyse
Eggenburg	egg01-1	Kalk	<input checked="" type="checkbox"/>
Unterstein	unt01-1	Kalksandstein	<input checked="" type="checkbox"/>
*			<input type="checkbox"/>

Datensatz: 1 von 2

Zwischen der Entwurfs- und Tabellenansicht kann man jederzeit über das Ansichts-Symbol ganz oben links wechseln.

Im Abfrage-Ergebnis können die Daten wie in der Original-Tabelle editiert werden.

Über *Datei-Exportieren* kann das angezeigte Ergebnis z.B. als Excell-Tabelle exportiert werden.

10.1. Platzhalter

Öffne die vorhin erstellte Abfrage *AbAnalyse* und speichere sie als neue Abfrage unter dem Namen *AbLokalit* ab.

Als Platzhalter für die Abfragekriterien können * und ? verwendet werden:

*E*enburg* oder *E??enburg* findet Eggenburg und würde auch Eckenburg finden.

auch **enburg* und **gg** findet Eggenburg

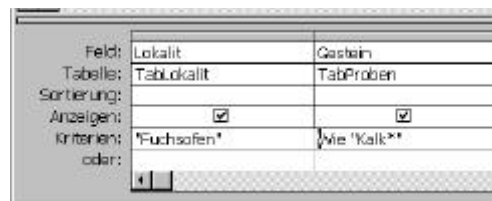
Wenn ich mit Platzhaltern arbeite, muß vor dem gesuchten Stichwort **Wie** stehen - siehe nachfolgende Beispiele.

10.2. Und/Oder Abfragen

Folgende Abfrage sucht nach Datensätzen, in denen **entweder** "Sand" im Feld *Gestein* vorkommt **oder** "Fuchsofen" im Feld *Lokalit*.



Folgende Abfrage zeigt die Proben vom Fuchsofen mit Kalk



Alle Proben mit Kalk oder Sand (durch die *Sternchen* wird auch z.B. "Kalksandstein" angezeigt)

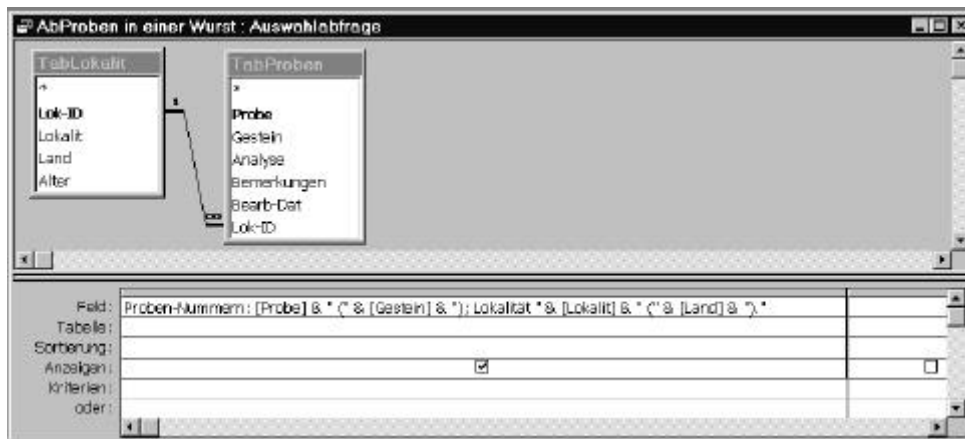
Feld:	Lokalität	Gestein
Tabella:	TabLokalität	TabProben
Sortierung:		
Anzeigen:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kriterien:		Wie "*"Kalk**
oder:		Wie "*"Sand**

10.3. 'Platzsparende' Abfrageergebnisse

Es können auch mehrere Felder in ein Feld geschrieben werden. Feldnamen werden durch eckige Klammern `[]` definiert. Weiters können beliebige Zeichen eingefügt werden, die durch `_` gekennzeichnet sind. Einzelne Felder und Zeichenfolgen werden durch `&` getrennt.

Beispiel:

Proben-Nummern: `[Probe] & " (" & [Gestein] & ") ; Lokalität " & [Lokalität] & " (" & [Land] & ") ."`



Abfrage speichern unter *AbProbenliste*

Das Abfrageergebnis:

AbProbenliste : Auswahlabfrage	
Proben-Nummern	
▶	kp01-1 (Mikritischer Kalk) ; Lokalität Fuchsofen (Kärnten).
	kp01-2 (Kalksandstein) ; Lokalität Fuchsofen (Kärnten).
	kp01-3 (Kalk) ; Lokalität Fuchsofen (Kärnten).
	unt01-1 (Kalksandstein) ; Lokalität Unterstein (Kärnten).
	unt01-2 (mikritischer Kalk) ; Lokalität Unterstein (Kärnten).
	egg01-1 (Kalk) ; Lokalität Eggenburg (Niederösterreich).
	egg01-3 (mikritischer Kalk) ; Lokalität Eggenburg (Niederösterreich).
*	

Datensatz: 1 von 7

10.4. Parameter-Abfragen

Schnellabfrage nach Lokalitäten:

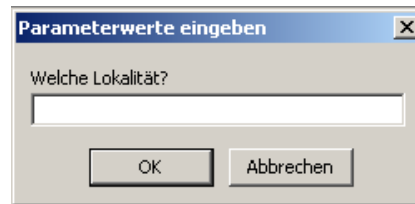
1.) Öffne die Abfrage *AbLokalität*.

Im Feld *Lokalität* schreiben wir in die Kriterien: **[Welche Lokalität?]**

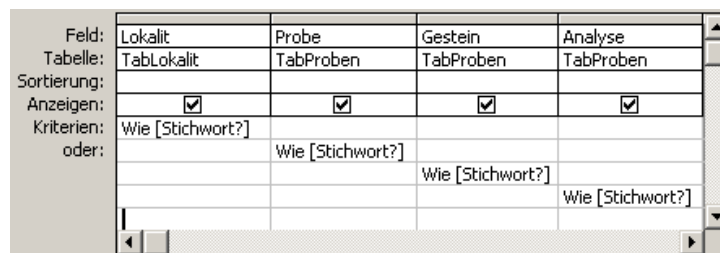
Um Platzhalter zu erlauben: **Wie [Welche Lokalität?]**

Abspeichern

2.) wird nun die Abfrage geöffnet, erscheint folgendes Fenster, in welchem ich meine Suchkriterien eingeben kann:

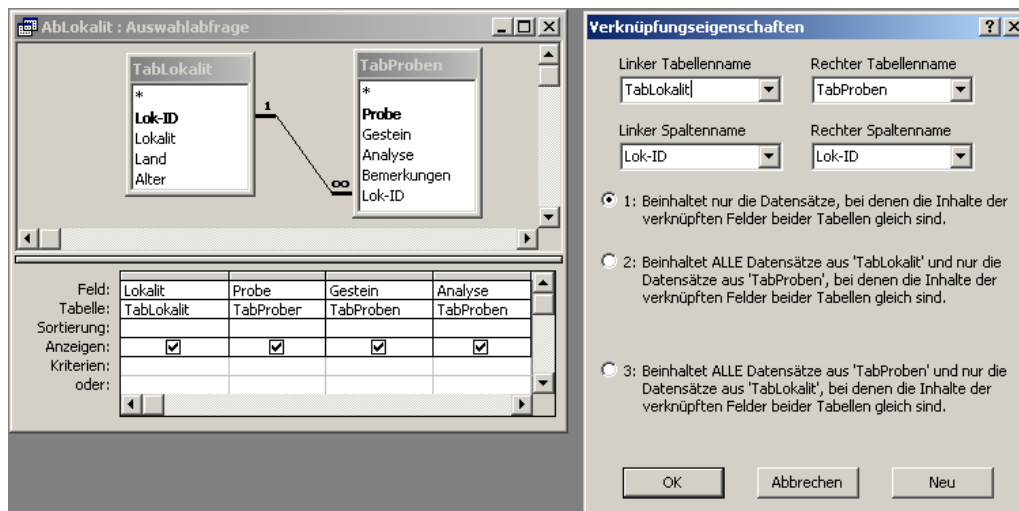


Oder: in jedes Feld *Wie [Stichwort]* als Oder-Abfrage eintragen, dann werden alle Felder durchsucht.



10.5. Verknüpfungstypen

Öffne *AbLokalit* in der Entwurfsansicht und klicke im oberen Fenster 2x auf die Beziehungslinie.



Einstellungen:

(1) = Exklusionsverknüpfung oder Inner Join

Konsequenz: In einer Abfrage werden nur jene Lokalitäten berücksichtigt, denen auch Proben zugewiesen wurden und - umgekehrt - nur jene Proben, die einer Lokalität zugewiesen wurden. Alle anderen Einträge scheinen in der Abfrage nicht auf. Daher 'Exklusionsverknüpfung', weil bei der Abfrage auf beiden Seiten (Tabelle "1" und Tabelle "N") Datensätze ausgeschlossen werden.

(2) = Links-Inklusionsverknüpfung oder Left Join

Konsequenz: In der Abfrage werden ALLE Lokalitäten aus TabLokalit (= Tabelle "1") angezeigt. Aus TabProben (=Tabelle "N") werden aber nur die Proben berücksichtigt, die einer Lokalität zugewiesen wurden.

(3) = Rechts-Inklusionsverknüpfung oder Right Join

Konsequenz: In der Abfrage werden alle Proben angezeigt. Es werden aber nur jene Lokalitäten angezeigt, zu denen es bereits Proben gibt.

Eselsbrücke zum Merken: im Beziehungsfenster stellen wir immer die Tabelle "1" (TabLokalit) auf die linke Seite, die Tabelle "N" (TabProben) auf die rechte Seite. 'Links-Inklusionsverknüpfung' bedeutet dann, dass alle Datensätze aus der linken Tabelle in die Abfrage inkludiert werden (und bei Rechts-Inklusionsverknüpfung eben umgekehrt).

Diese Einstellung kann ich bereits bei der Definition der Beziehungen einstellen (siehe Kapitel **Beziehungen definieren**), indem ich im Fenster **Beziehungen bearbeiten** auf den Knopf **Verknüpfungstyp** klicke. Unabhängig von dieser Voreinstellung kann ich dies aber für jede Abfrage individuell einstellen.

10.6. Ein paar Worte zu SQL

Um die Abfrage von verschiedenen Datenbanken zu vereinheitlichen, wurde eine standardisierte Abfragesprache entwickelt: SQL (für Structured Query Language). Nahezu alle relationalen Datenbanken verfügen heute über SQL-Fähigkeit.

SQL ist eine produkt(=software)unabhängige und standardisierte Abfragesprache, mit der Datenbanken verwaltet und ausgewertet werden könne. Grundsätzlich kann man immer statt eines Tabellen-, Abfragen- oder Feldnamen eine SQL-Anweisungen eingeben.

Z.B. stellen Abfragen in Access, die verschiedene Tabellen einbeziehen, eine SQL-Anweisung dar.

Ein Beispiel können wir sehen, indem wir *AbAnalyse* in der Entwurfsansicht öffnen und dann links oben im Ansichtssymbol *SQL-Ansicht* wählen.

Der SQL-Code sieht so aus:

```
SELECT TabLokalit.Lokalit, TabProben.Probe, TabProben.Gestein, TabProben.Analyse
FROM TabLokalit INNER JOIN TabProben ON TabLokalit.[Lok-ID] = TabProben.[Lok-ID]
WHERE ((TabProben.Analyse)=Yes);
```

SELECT bedeutet, dass gezielte Felder aus Tabellen selektiert werden. Der "." bedeutet z.B., dass das Feld *Lokalit* zur Tabelle *TabLokalit* gehört. Dies sind die Tabellen und Felder, die im oberen Fenster angezeigt werden. FROM stellt dann den Bezug auf eine Tabelle her; danach kommt die Tabelle, aus der die Datensätze entnommen werden soll. WHERE ist dann die Abfrageoption; in diesem Falle jene Datensätze, in denen *Analyse* angeklickt wurde

10.7. SQL Union-Abfragen

Erstellen eine Vereinigungsabfrage: Es werden in den Abfrageergebnissen Felder aus einer oder mehreren Tabellen oder Abfragen in einem Feld oder einer Spalte kombiniert.

PROBLEM: Ich will wissen, an welchen Lokalitäten Kalk vorkommt

und erstelle daher folgende Abfrage namens *AbLokalitProbe*:



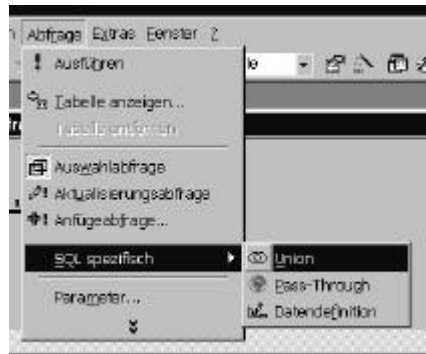
Das Ergebnis sieht dann z.B. so aus:

	Lokalit	Gestein
▶	Fuchsofen	Kalkstein
▶	Fuchsofen	Kalk Sand
▶	Unterstein	Kalk
*		

UNBEFRIEDIGEND: Ich erhalte eine Tabelle, in der einzelne Lokalitäten unnötig oft vorkommen! Bei großen Datenmengen ist das eine unnötige Redundanz, weil ich ja einfach nur wissen wollte, wo Kalk vorkommt!

LÖSUNG: SQL Union-Abfrage

Erstelle eine neue Abfrage im Entwurfsmodus. Dann wähle im Menü *Abfrage -> SQL spezifisch -> Union*.



In das Fenster schreibe ich folgenden Code:

```
SELECT Lokalit FROM AbLokalitProbe  
UNION SELECT Lokalit FROM AbLokalitProbe;
```

Speichern unter *AbUnionSelect*

und erhalte folgendes Ergebnis:

Lokalit
Fuchsofen
Unterstein

Datensatz: 1 2

11. Ein Formular erstellen

Ein Formular ist eine höchst individuelle Form der Tabellendarstellung. Sie bestimmen, welche Felder darin angezeigt werden, können diese beliebig arrangieren sowie Grafiken zeichnen und einfügen.

11.1. Einfache Formulare aus einer Tabelle

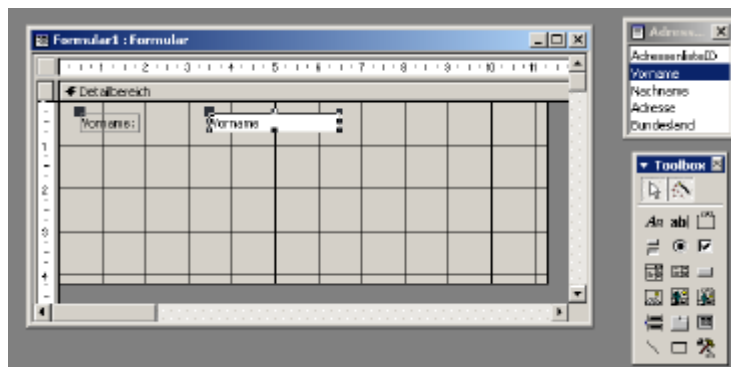
Ohne Formular-Assistenten

Wir schließen zunächst alle Tabellen und Formulare und erstellen wieder aus der im **Beispiel 1** erstellten **Adressen-Datenbank** ein neues Formular.


Markieren Sie *Formulare* und wählen Sie dann *Neu - Entwurfsansicht*. Wählen Sie im Auswahlfenster die gewünschte Tabelle aus.

Sie erhalten nun ein leeres Formular-Blatt, in welches die gewünschten Felder eingefügt werden können.

Die vorhandenen Felder der vorhin ausgewählten Tabelle können nun aus dem kleinen separaten Fenster markiert und in das Feld gezogen werden.



Der Feldname erscheint immer links vom Feldeintrag. Dieser Feldname kann unabhängig vom eigentlichen Feld verschoben werden.

Um zu sehen, wie das Formular dann schließlich aussehen wird, klicken Sie wieder ganz links oben auf den 'Ansicht'-Button .

11.2. Formulare aus verknüpfte Tabellen erstellen

Formulare für komplexere Datenbanken (z.B. 1:N Beziehungen) können nun auf zwei Arten erstellt werden: (1) Die Haupttabelle (Tabelle "1") bildet das Hauptformular und die Nebentabelle (Tabelle "N") bildet darin ein Unterformular - vergleichbar dem oben erläuterten Unterdatenblatt in der Haupttabelle (Tabellenansicht); (2) es werden einfach zwei verschiedene Formulare aus den zwei vorhandenen Tabellen angefertigt, die dann miteinander verknüpft sind.

11.2.1. Verknüpfte Formulare

Anhand von **Beispiel 2**.

Wir öffnen die Datei **Proben** und wählen *Formular - Neu - Formular-Assistent*

Einfügen von Feldern aus verschiedenen Tabellen. Daher jeweils eine Tabelle bei *Tabellen/Abfragen* auswählen und die gewünschten Felder in dem rechten Feld einfügen.

D.h., aus *TabLokalit* und *TabProben* werden alle Felder ausgenommen *Lok-ID* eingefügt (denn *Lok-ID* ist nur der Primärschlüssel, den wir ja nicht selber eingeben und dessen Wert uns nicht weiter interessiert).

Dann auf *Weiter* klicken.

Access erkennt nun, dass es sich um verknüpfte Tabellen handelt und fragt uns, ob wir *Formulare mit Unterformular(en)* oder *Verknüpfte Formulare* anlegen wollen.

Für unser erstes Formular wählen wir *Verknüpfte Formulare*.

Wenn man wieder *Weiter* klickt, kann man einen Hintergrund auswählen.

Als nächstes werden wir nach den Formular-Titeln gefragt und wählen wieder aussagekräftige Namen, damit spätere Manipulationen übersichtlicher werden.

Für das 1. *Formular* (= Hauptformular, entspr. Tabelle "1"): *FmLokalit1*

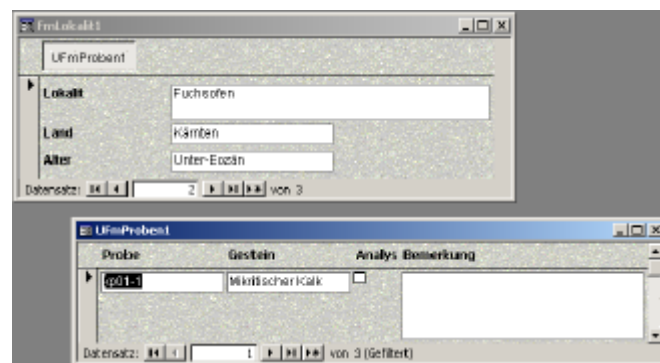
Für das 2. *Formular* (= Unterformular, entspr. Tabelle "N"): *UFmProben1*

Wir wählen den Zusatz "1", weil wir noch weitere Formulare erstellen werden

Nun wählen wir *Das Hauptformular öffnen* und *Fertig stellen*.

Wir erhalten nun das Formular *FmLokalit*. Das verknüpfte Unterformular *UFmProben* können wir öffnen, indem wir auf den Knopf *UFmProben* klicken.

Nun öffnet sich unser Unterformular mit den Proben. Wenn wir nun im *FmLokalit* in der unteren Navigationsleiste zur nächsten Lokalität gehen, ändert sich auch entsprechend der Inhalt im verknüpften Formular



11.2.2. Haupt- und Unterformular

Nun machen wir das selbe mit Haupt- und Unterformular am selben Beispiel.

Schließen Sie alle offenen Tabellen und Formulare.

Die ersten Schritte sind gleich wie oben

Wie vorher wählen Sie *Formular - Neu - Formular-Assistent*

Einfügen von Feldern aus verschiedenen Tabellen. Daher jeweils eine Tabelle bei *Tabellen/Abfragen* auswählen und die gewünschten Felder in dem rechten Feld einfügen.

D.h., aus *TabLokalit* und *TabProben* werden alle Felder ausgenommen *Lok-ID* eingefügt (denn *Lok-ID* ist nur der Primärschlüssel, den wir ja nicht selber eingeben und dessen Wert uns nicht weiter interessiert).

Dann auf *Weiter* klicken.

Access erkennt nun, dass es sich um verknüpfte Tabellen handelt und fragt uns, ob wir *Formulare mit Unterformular(en)* oder *Verknüpfte Formulare* anlegen wollen.

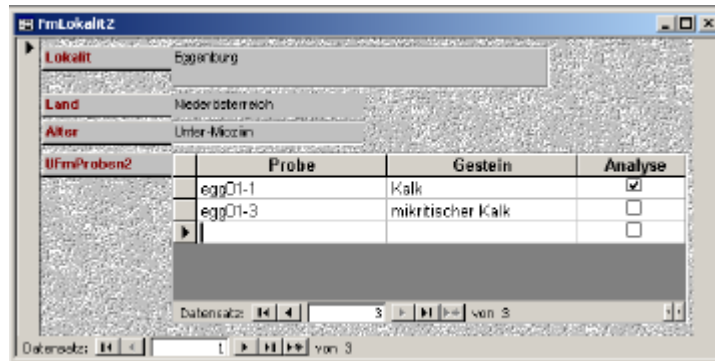
Im Gegensatz zu vorher wählen wir nun *Formulare mit Unterformular(en)*.

Als Layout wählen wir *Datenblatt*, wählen dann wieder einen Hintergrund aus und schließlich die Namen der Formulare.

In Anlehnung an die vorhergehenden Formulare wählen wir als Formular-Namen:


Für das 1. *Formular* (= Hauptformular, entspr. Tabelle "1"): *FmLokalit2*

Für das 2. *Formular* (= Unterformular, entspr. Tabelle "N"): *UFmProben2*



Das Unterformular wird als Tabelle angezeigt, weil wir im Assistenten *Datenblatt* ausgewählt haben.

11.2.3. Layout des Formulars verändern

In den Layout-Entwurfsmodus kommen wir wieder mit 

Bei der Änderung des Layouts beachten Sie, dass Sie das Unterformular separat vom Hauptformular editieren müssen.

Schließen Sie das Formular, dann sehen Sie im Fenster das Hauptformular *FmLokalit2* und das Unterformular *UFmProben2*.

Beachten Sie, dass *UFmProben2* nur als Tabellenform, nicht als eigentliches Formular angezeigt wird (dies haben wir dem Assistenten angegeben). Wenn Sie also z.B. die Spaltenbreite von *UFmProben2* verändern wollen, müssen Sie das über den Tabellenentwurf machen!

Layout des Hauptformulars: Markieren Sie *FmLokalit2* und wählen Sie *Entwurf*. Das Unterformular wird im Entwurf angezeigt - aber die eigentliche Darstellungsform wird dann eine Tabelle sein.

Vergrößern und verschieben: Wenn Sie ein Objekt anklicken und die Maustaste gedrückt halten, wird der Mauscursor zur Hand und Sie können das Objekt verschieben. Wenn Sie einmal draufklicken und die Maus loslassen, erhalten Sie schwarze Rechtecke rund um das Objekt und können die Größe verändern.

Text: Einmal mit der Maus auf das Feld klicken. Sie können nun den Text markieren und Formatieren (Farbe, Schriftart ...) über die Menü-Kopfzeile bearbeiten.

Bilder (z.B. gescannte) könne Sie über *Einfügen - Grafik* einfügen

In der Formular - Entwurfsansicht erhalten sie folgende 'Toolbox':



Optionen der Toolbox:

Objekte markieren: auswählen von Menübefehlen, Objekten, etc. im Formular; verschieben, vergrößern, etc.

Steuerelement-Ass.: ist dieser Knopf aktiviert, erscheint beim Einfügen eines neuen Elementes (s.u.) ein Assistent für die Formatierung des Elementes

Bezeichnungsfeld:	erzeugt ein Feld mit rein beschreibendem Text, wie Überschriften oder Feldbeschriftungen
Textfeld:	Erzeugt neue Eingabefelder
Optionsgruppe:	Optionsgruppen können Kontrollkästchen, Optionsfelder oder Umschaltflächen (s.u.) zusammenfassen. Innerhalb einer Gruppe kann lediglich ein Wert ausgewählt werden.
Umschaltfläche:	können z.B. zur Eingabe von Ja/Nein Werten verwendet werden.
Optionsfeld:	wie Umschaltflächen, aber optisch unterschiedlich
Kontrollkästchen:	ebenfalls wie Umschaltflächen, aber optisch unterschiedlich.
Kombinationsfeld:	erzeugt ein Feld mit einer Auswahlliste, die durch Anklicken des Pfeils aktiviert wird.
Listenfeld:	Auswahl eines Wertes aus einer vordefinierten Liste.
Befehlsschaltfläche:	aktiviert einen Befehl, z.B. ein Makro
Bild:	Einfügen eines Bildes (Pixel-Grafik) am Formular (das Bild ist dann nicht an einen einzelnen Datensatz gebunden).
Objektfeld:	ungebundene Objekte wie Grafiken. Das Objekt wird nicht innerhalb der Datenbankdatei gespeichert (daher 'ungebunden')! Im Gegensatz zu 'Bild' können auch Tabellen udgl. eingefügt werden.
Gebundenes Objektf.:	Objekt w.o., aber das Objekt wird dann in der Datenbank gespeichert.
Seitenwechsel:	Seitenwechsel im Formular, falls das Formular für den Bildschirm zu groß wird.
Register-Steuerel.:	erstellt Formular in Form eines Registers, falls das Formular für den Bildschirm zu groß wird.
Unterformular/Ber.:	Einfügen eines Unterformulars
Linie:	Einfügen von Linien
Rechteck:	Einfügen von Rechtecken
Weitere Steuerel.:	Zeigt weitere Steuerelemente an (z.B. Kalender-Steuerelement).

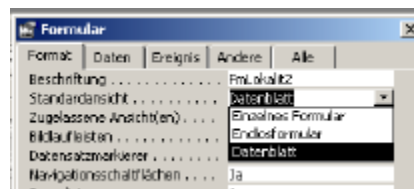
11.2.4. Darstellungsformen des Unterformulars


Über das Symbol  öffnen Sie das Fenster **Eigenschaften**.

Wenn Sie nun irgendein Objekt des Formulars anklicken, sehen Sie dessen Eigenschaften.

Klicken Sie nun 2x auf die linke obere Ecke des **Unterformulars**.

Im Register **Format** des Eigenschaftfensters können Sie zwischen drei verschiedenen Formatierungen wählen.



Über das Ansichtssymbol  sehen Sie dann, wie die unterschiedlichen Darstellungsformen des Unterformulars aussehen:

Datenblatt:

	Probe	Gestein	Analyse
▶	egg01-1	Kalk	<input checked="" type="checkbox"/>
	egg01-3	mikritischer Kalk	<input type="checkbox"/>
*			<input type="checkbox"/>

Datensatz: 1 von 2

Einzelnes Formular:

Endlosformular:

11.2.5. Das Eigenschaftfenster

Je nachdem, welches Objekt im Formular angeklickt ist, werden unterschiedliche Eigenschaften angezeigt.

Hinweise zur Bedeutung der einzelnen Formatierungskriterien erhalten Sie, indem Sie in das jeweilige Feld klicken und auf der Tastatur die Steuertaste **F1** drücken.

11.2.6. Kombinationsfelder

Kombinationsfelder sind 'Auswahlfenster' im Formular, aus denen ich Werte auswählen kann.

Auswahlfenster mit Assistenten erstellen

Beispiel: In unserer Datenbank **Proben.mdb** haben wir das Feld **Alter**. Dieses Feld enthält chronostratigraphische Begriffe, die genormt sind. Wenn man das Alter zu jedem Datensatz 'händisch' eingibt, kann man sich leicht verschreiben. Außerdem ist das Schreiben zeitaufwendiger.

Lösung: Wir erstellen eine neue Tabelle und nennen diese **TabChrono**. Diese Tabelle enthält nur ein Feld, nämlich den chronostratigraphischen Begriff; dieses nenne wir **Chrono**. Nachdem jeder Begriff nur einmal vorkommt, verwenden wir dieses Feld auch gleichzeitig als *Primärschlüssel*. In der Tabellenansicht geben wir nun die chronostratigraphischen Einheiten (Unter-Eozän, Mittel-Eozän, Unter-Miozän, etc.) ein.

Dann öffnen wir unser Formular **FmLokalit2** im *Entwurfsmodus*. Wir könnten jetzt das vorhandene Feld **Alter** entsprechend modifizieren. Um es einfacher zu machen, benutzen wir aber den Assistenten und löschen zunächst das vorhandene Feld **Alter**. (Dies wirkt sich NICHT auf die Tabelle aus, d.h., das Feld wird nicht in der Tabelle gelöscht!).

Nun klicken wir in der **Toolbox** auf das Symbol für das **Kombinationsfeld** und klicken auf der Maske dorthin, wo es dann erscheinen soll. Wir erhalten automatisch den Assistenten zum Formatieren dieses Feldes.

Im Assistenten wählen wir die erste Option, weil wir die Werte ja aus der vorhandenen Tabelle **TabChrono** entnehmen wollen. Dies geben wir im dann im nächsten Fenster auch an. Im nächsten Fenster bringen wir das Feld **Chrono** (wir haben ja nur das eine in dieser Tabelle) in die *Ausgewählten Felder*. Später werden wir gefragt, wo wir die chronostrat. Einheiten, die wir aus der Tabelle auswählen speichern wollen. Logischerweise wählen wir hier das Feld **Alter** aus unserer Tabelle **TabLokalit**. Schließlich müssen wir das Kombinationsfeld an sich noch benennen und geben ihm den Namen **Alter**.

Wenn wir in die andere Ansicht wechseln, können wir nun weitere Daten eingeben und brauchen immer nur das Alter aus dem Auswahlfenster auswählen.

Beachte: wenn ich eine neue chronostrat. Einheit einfügen will, muss ich dies in der Tabelle **TbChrono** machen.

Auswahlfenster 'manuell' erstellen

Zur Erstellung von Kombinationsfenstern muß man aber nicht das vorhandene Feld aus dem Formular löschen, sondern man kann auch ein vorhandenes Textfeld einfach in ein Kombinationsfeld umwandeln.

Als nächstes Kombinationsfeld in unserem Beispiel bietet sich das Feld **Land** an.

Wir erstellen wie vorher eine neue Tabelle namens **TabLand**, wieder mit nur einem Feld **Land**, das wir als Primärschlüssel definieren.

Geben Sie nun die Länder ein.

Dazu klicke ich mit der rechten Maustaste auf das Feld und wähle *Ändern zu - Kombinationsfeld*.

Dann wählen wir im *Eigenschaften-Fenster* im Register **Daten** bei **Datensatzherkunft** die Tabelle **TabLand** aus.

Übung: Geben Sie nun im Hauptformular eine neue Lokalität ein, z.B.:

Lokalität: Haunsberg, Land: Salzburg, Alter: Unter-Eozän

Kombinationsfelder in die ich auch Daten eingeben kann

Für unser Feld Land wollen wir ein Kombinationsfeld erstellen, aus dem ich vorhandene Einträge auswählen kann (ähnlich wie beim Alter durchgeführt), in das ich aber auch direkt neue Daten zufügen kann ohne die Tabelle extra zu öffnen.

- 1.) Abfrage erstellen namens **AbNurLand**. In diese nur ein Feld, nämlich **Land**, einfügen.
- 2.) Neues Kombinationsfeld in Formular einfügen und bei den Eigenschaften folgendes ändern:

Steuerelementinhalt:	Land
Herkunftstyp:	Tabelle/Abfrage
Datensatzherkunft:	AbNurLand
Nur Listeneinträge:	Nein
Automatische ergänzen:	Ja

Kombinationsfenster zur Datennavigation

Beispiel: Ein Formular dient ja nicht nur zur Dateneingabe, sondern auch zur übersichtlichen Darstellung der Daten. Die Navigation zwischen den Datensätzen ist aber in der bisherigen Form unübersichtlich.

Lösung: Auch dieses Problem kann man mit einem Kombinationsfeld in Form eines Navigationsfeldes lösen. Dazu wählen wir in der Entwurfsansicht des Formulars wieder *Kombinationsfeld* und klicken am Formular auf die entspr. Stelle. Im Assistenten wähle ich jetzt aber die **dritte Option** und wähle die Felder, die im Navigationsfenster angezeigt werden sollen. Wir wählen hier alle, ausgenommen **Lok-ID**. Schließlich benenne ich das Kombinationsfeld als **Navigation**. Wenn ich jetzt wieder in die Normalansicht gehe, kann ich dieses Fenster anklicken und sehe dann die vorhandenen Datensätze. Wähle ich einen Datensatz aus, wird er mir angezeigt.

Beachte: Dieses Navigationsfenster dient nicht der Dateneingabe!!!

Übung: Formatiere das Formular-Layout und arrangiere die Felder.

11.2.7. Befehlsschaltflächen

Eine andere Art der Navigation ist über die Befehlsschaltflächen möglich. Dazu in der Formular - Entwurfsansicht aus der Toolbox die Option **Befehlsschaltflächen** auswählen und im Formular platzieren.


Folge dann den Anweisungen.

Übung: Verschaffe dir einen Überblick über die Optionen und gestalte weitere Schaltflächen für die Navigation (z. B. Suchen, löschen, etc.).

11.3. Formulare mit 'platzsparender' Darstellung für das Durchblättern von Datensätzen

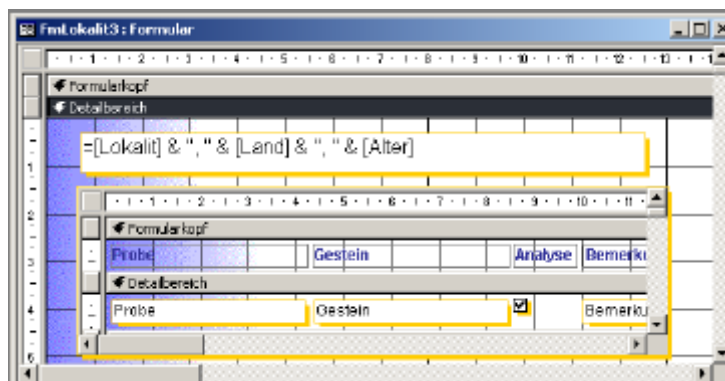
Die einzelnen Felder eines Datensatzes können auch in einem einzelnen Feld dargestellt werden.


Öffne *FmLokalit2* in der Entwurfsansicht, wähle im Menü *Datei - Speichern unter* und erstelle so ein neues Formular namens *FmLokalit3*.

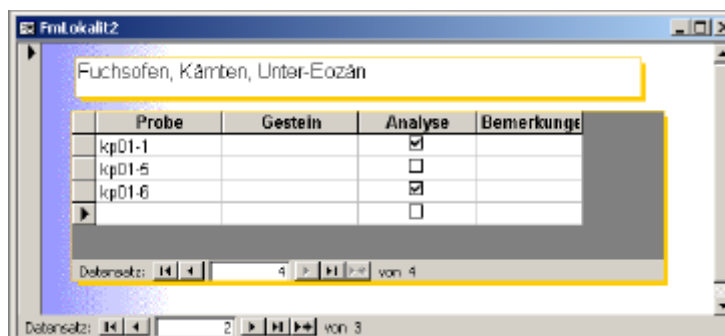
In der *Toolbox* klicke auf Textfeld  und platziere ein neues Textfeld im Hauptformular. Dies ist nun ein **ungebundenes** Textfeld, d.h., dieses Feld kommt nicht aus der Tabelle

Lösche nun aus dem Hauptformular die Felder *Lokalit*, *Land* und *Alter*.

In das neue Textfeld schreibe nun: `= [Lokalit] & ", " & [Land] & ", " & [Alter]`



wechsle nun mit  in die Datenblatt-Ansicht



So kann ich ein Formular für die Dateneingabe erstellen und ein zweites für das Durchblättern der Daten. Ich habe somit nicht nur eine platzsparende Darstellung, sondern auch Datensicherheit, da die Daten in diesem zweiten Formular nicht verändert werden können.

Dies ist natürlich auch für das Unterformular möglich

12. Berichte

Berichte sind eine für das Drucken vorgesehene Darstellungsform von Tabellen und Abfragen

12.1. Berichte aus Tabellen

Berichte - Neu - Berichtsassistent

Bei der Auswahl der Tabellen und Felder gehen wir wieder wie bei der Erstellung des Formulars vor:

Einfügen von Feldern aus verschiedenen Tabellen. Daher jeweils eine Tabelle bei *Tabellen/Abfragen* auswählen und die gewünschten Felder in dem rechten Feld einfügen.

D.h., aus *TabLokalit* und *TabProben* werden alle Felder ausgenommen *Lok-ID* eingefügt (denn *Lok-ID* ist nur der Primärschlüssel, den wir ja nicht selber eingeben und dessen Wert uns nicht weiter interessiert).

Dann auf *Weiter* klicken.

Der Assistent erkennt wieder, dass wir es mit zwei verknüpften Tabellen zu tun haben. Daher fragt er nun, ob die Daten gruppiert werden sollen.

Ohne Gruppierung

Wenn ich *nach TabProben* auswähle, sind die Daten nicht gruppiert. Dieses versuchen wir als erstes.

Wir klicken *Weiter* und fügen **keine** vorgeschlagene Gruppierungsebene ein. Stattdessen klicken wir wieder *Weiter*.

Nun kann ich wählen, ob die Daten sortiert werden sollen und im nächsten Fenster kann ich die Anordnung der Daten festlegen.

Als Berichts-Name wählen wir **BerLokalitUngruppiert** (Leerzeichen und längere Namen wären auch möglich).

Als Ergebnis sehen wir den Bericht, den ich nun ausdrucken kann.

Da wir keine Gruppierung ausgewählt haben, kommt es zu **Datenredundanzen!**

Daher versuchen wir es jetzt mit Gruppierung:

Mit Gruppierung

Wir starten den Assistenten wieder und geben die selben Felder wie im vorigen Durchgang an.

Diesmal wählen wir allerdings die Gruppierung, indem wir *nach TabLokalit* auswählen.

Im Vorschau-Fenster sehe ich die Gruppierung.

Als Berichts-Namen geben wir **BerLokalitGruppiert** an und sehen dann, wie unsere Daten gruppiert wurden.

Übung: Formatiere den Bericht.

'Platzsparende' Berichte

Ähnlich wie zuletzt beim Formular ausgeführt, kann ich auch in Berichten ein Feld einfügen, in welches mehrere einzelne Felder geschrieben werden.

12.2. Berichte aus Abfragen

Wenn man im Bericht nur ausgewählte Datensätze anzeigen will, kann man einen Bericht basierend auf einer Abfrage erstellen.

Wie vorhin wählen wir den Berichtsassistenten, wählen aber diesmal unter *Tabellen/Abfragen* unsere vorher angefertigte Abfrage *AbAnalyse* aus.

D.h., in meinem Bericht scheinen nur jene Proben auf, die ich bereits analysiert habe!

Alle Felder einfügen und weiter klicken.

Nach den bereits bekannten Auswahlkriterien wählen wir den Dateinamen *BerAnalyse*.

13. Makros

Ein Makro kann eine Folge von Aktionen, die Sie in Ihrer Datenbank durchführen möchten, unter einem Namen speichern und auf Knopfdruck ausführen.

Z.B.: Wenn Sie im Lokalitäten-Hauptformular Daten eingeben, möchten Sie wissen, welche Proben Sie überhaupt schon analysiert haben. Normalerweise müssten Sie dazu aus dem Hauptformular aussteigen, im Hauptmenü *Abfragen* auswählen und dann die Abfrage *AbAnalyse* öffnen.

Stattdessen können Sie ein entspr. Makro schreiben, das Sie dann vom Hauptformular aus durch Mausclick aktivieren und das diese Abfrage automatisch öffnet.

Dazu wählt man in den 'Objekten' - 'Makros' und *Neu*. Bei *Aktion* wählt man *Öffnen Abfrage*, bei *Abfragename* wählt man *AbAnalyse*, bei *Ansicht* wählt man *Datenblatt* und bei *Datenmodus* - *Bearbeiten*. Dieses Makro speichern wir dann unter *OpenAbAnalyse* ab.

Dann öffnet man das Formular *FmLokalit2* in der Entwurfsansicht. Jetzt wählen Sie aus der 'Toolbox' die Option Befehlsschaltfläche (die dann das Makro aktivieren wird) und klicken am Formular dort hin, wo diese Schaltfläche platziert sein soll. Im erscheinenden Fenster wählen Sie *Diverses* und *Makro ausführen* und wählen dann das eben erstellte Makro *OpenAbAnalyse* aus. Wenn Sie *Weiter* klicken, können Sie die Schaltfläche formatieren. Wählen Sie *Text* und schreiben Sie z.B. *Analysierte Proben*. Dieser Text wird dann auf der Schaltfläche erscheinen. Zuletzt müssen Sie noch die Schaltfläche benennen (dieser Name wird im Formular nicht aufscheinen). Wählen Sie als prägnanten Namen *MkAbAnalyse* (Mk steht für Makro).

Übung: Als zweites Makro werden wir eine Bearbeitungsmöglichkeit der chronostratigraphischen Tabelle einfügen. Wenn man bei der Probeneingabe im Formular *FmLokalit2* feststellt, dass das gewünschte Alter noch nicht vorhanden ist, müsste man die Tabelle über das Hauptmenü öffnen. Wir fügen jetzt am Formular ein Makro ein, mit dem die Tabelle *TabChrono* geöffnet wird, damit ich dieser Tabelle einen neuen Begriff zufügen kann.

14. Übersichtsmanager

Der Übersichtsmanager ist die einfachste und schnellste Methode zur Erstellung einer Menüsteuerung beim Öffnen einer Datenbank. Das Ziel des Übersichtsmanager ist es, dem Benutzer die Arbeit mit der Datenbank zu erleichtern.

Das Grundelement für diese Art der 'Benutzerführung' sind verschiedene 'Übersichten'.

Übersichtsmanager anlegen

Wir beginnen mit der Erstellung der **Hauptübersicht**. In der Hauptübersicht werden dann beim Start der Datenbank die einzelnen Menüpunkte angeführt.

Wählen Sie im Menü *Extras* -> *Datensatz-Dienstprogramme* -> *Übersichtsmanager* aus.

Wir werden nun darauf hingewiesen, dass noch keine Übersicht existiert und wählen 'Ja'

Wir erhalten nun ein Fenster mit *Main Switchboard (Default)* und wählen *Bearbeiten*.

Im nächsten Fenster können wir die **Menüpunkte** durch klicken von 'Neu' hinzufügen:

Als erste Menüoption wollen wir das Starten unseres Hauptformulars für die Eingabe von Daten angeben.

1. Dazu geben wir bei *Text* folgendes ein: 'Hauptformular Lokalitäten starten'.
2. Bei *Befehl* geben wir 'Formular im Bearbeitungsmodus öffnen ' aus
3. Bei *Formular* geben wir nun 'FmLokalit2' an; dieses Formular wird also beim Auswählen der Menüoption gestartet.

Nun klicken wir 'OK' und schließen die beiden verbleibenden Fenster.

Übersichtsmanager starten

Unter den Tabellen finden wir jetzt eine neue Tabelle mit dem Namen *Switchboard Items*. Diese Tabelle enthält die Menüoptionen. Außerdem haben wir jetzt ein neues Formular namens *Switchboard*.

Wenn wir das Formular *Switchboard* aufrufen, erhalten wir unsere Hauptübersicht.

Nachdem wir nur eine Menüoption ausgewählt haben, erscheint auch nur die Option 'Hauptformular Lokalitäten starten'. Wenn wir diese Option anklicken, erscheint auch das entspr. Formular und ich kann Daten abfragen.

Übersichtsmanager beim Aufrufen der DB automatisch starten

Damit der Übersichtsmanager automatisch beim Öffnen der Datei geöffnet wird, wähle ich im Menü *Extras* -> *Start* aus und wählen Sie das Formular *Switchboard*.

Schließen Sie nun die Datei *Proben* (z.B. durch *Datei* -> *Schließen*) und öffnen Sie sie wieder. Nun erscheint automatisch unser Übersichtsmanager.

Weitere Menüoptionen einfügen

Wählen Sie wieder im Menü *Extras* -> *Datensatz-Dienstprogramme* -> *Übersichtsmanager* aus. Wählen Sie dann *Bearbeiten*, so erhalten Sie die eine vorhandene Menüoption.

Um neue Menüoptionen einzufügen, wählen Sie 'Neu'.

Um nun z.B. einen Bericht über die bereits analysierten Daten anzeigen zu lassen, wählen Sie *Befehl - Bericht öffnen* und *Bericht - BerAnalyse*. Benennen Sie die Option z.B. mit *Bericht über analysierte Daten*.

'Tabelle öffnen' als Makro einfügen

Die Option, eine Abfrage zu öffnen, ist hier nicht vorgesehen. Dafür können wir unser vorher angelegtes Makro verwenden.

Wir gehen wieder in den Übersichts-Manager, wählen *Bearbeiten* und wählen ein neues *Element* mit *Neu*.

Wir benennen diese Menüoption mit *Abfrage analysierter Proben anzeigen*, wählen dann *Makro ausführen* und schließlich unser Makro *OpenAbAnalyse*.

15. N:M Beziehung

Grundsätzlich ist es möglich, dass ein Datensatz aus einer Tabelle N mehrere passende Datensätze der Tabelle M und umgekehrt besitzt. Um Datenredundanzen zu vermeiden, wird man hier eine N:M Beziehung einführen.

In relationalen Datenbanken ist eine derartige Beziehung nicht direkt herzustellen. Daher muss eine weitere Tabelle Z eingefügt werden, die dafür sorgt, dass die vorliegende N:M Beziehung in zwei 1:N Beziehungen aufgespalten wird. D.h., die Tabelle Z beinhaltet die Verknüpfung zwischen Tabelle N und M.

Tabelle N zu Z sowie Tabelle M zu Z stellen dann jeweils eine 1:N Beziehung dar.

Ähnlich wie bei der oben durchgeführten 1:N Beziehung werden die Primärschlüssel von Tabelle N und M in Tabelle Z eingefügt.

Beispiel 3

Erstelle eine Datenbank für Aufschlüsse und deren Koordinaten sowie mit den vorkommenden lithostratigraphischen Einheiten und deren Gesteinscharakterisierung. Füge außerdem zu jedem Datensatz eine Kartenskizze als Bild ein.

Aufschluss Barmstein, RW (Rechtswert) 430963 HW (Hochwert) 283448, Salzburg (Karte: barmstein.jpg).
Lithostratigraphie: Barmsteinkalk und Oberalmerschichten

Aufschluss Trattberg, RW 445621 HW 278038, Salzburg (Karte trattberg.jpg).
Lithostratigraphie: Barmsteinkalk und Oberalmerschichten

Aufschluss Tressenstein, RW 484303 HW 276538, Steiermark (Karte tressenstein.jpg).
Lithostratigraphie: Tressensteinkalk, Plassenkalk und Oberalmerschichten

Aufschluss Plassen, RW 470632 HW 270277, Oberösterreich (Karte plassen.jpg).
Lithostratigraphie: Plassenkalk

Alter und Charakterisierung der Gesteine:

Oberalmerschichten: Tithonium bis Berriasium; bankige mikritische Kalke

Barmsteinkalk: Tithonium bis Berriasium; grobdetritische Kalkbänke, eingeschaltet in die Oberalmerschichten

Tressensteinkalk: Kimmeridgium bis Tithonium; massige Kalkbrekzien mit Komponenten des Plassenkalkes

Plassenkalk: Kimmeridgium bis Berriasium; massiger Riffkalk

Übung: Überlege dir die Datenbankstruktur. Welche Beziehungen bestehen zwischen den Angaben, wie können Datenredundanzen vermieden werden?

Tabellen erstellen:

Achten Sie zunächst darauf, dass die drei vorhandenen Karten in Ihrem Arbeitsverzeichnis liegen!

Nun werden die Tabellen erstellt:

1. Tabelle **TabAufschluss** mit den Feldern:

<i>Aufschluss</i>	für den Namen des Aufschlusses. Textfeld, sicherheitshalber 255 Zeichen. Indiziert: <i>ja</i> (<i>Ohne Duplikate</i>). Eingabe erforderlich: <i>ja</i> . Der Name des Aufschlusses sollte jeweils einzigartig sein und kann daher als <i>Primärschlüssel</i> verwendet werden
<i>RW</i>	Rechtswert. Feldgröße: <i>LongInteger</i> , Gültigkeitsregel: "??????" (d.h., immer 6-stellig) (es wäre auch ">=999999" möglich); Gültigkeitsmeldung: <i>Achtung: RW immer 6-stellig!</i>
<i>HW</i>	Hochwert. Rest wie oben.
<i>Land</i>	
<i>Foto</i>	für die Abbildung der Karte. Felddatentyp: <i>OLE-Objekt</i>
<i>Bemerkung</i>	<i>Memo</i> - Feld

2. Tabelle *TabLitho* mit den Feldern:

- Litho* Lithostratigraphische Einheit. Kann wegen der Einzigartigkeit als *Primärschlüssel* verwendet werden. Eingabe erforderlich: *Ja*.
- Gestein* Gesteinscharakterisierung, sicherheitshalber ein *Memo* - Feld
- Alter* Könnte als eigene Tabelle angefertigt werden. Wir verwenden hier aber ein einfaches Textfeld.

3. Tabelle *Auf-Litho* als Verknüpfungsdatei mit den beiden Primärschlüssel-Feldern aus den anderen Tabellen:

Dazu markiert man in den anderen beiden Feldern jeweils die **ganze Zeile** des Primärfeldes, kopiert es, und fügt es in *Auf-Litho* ein

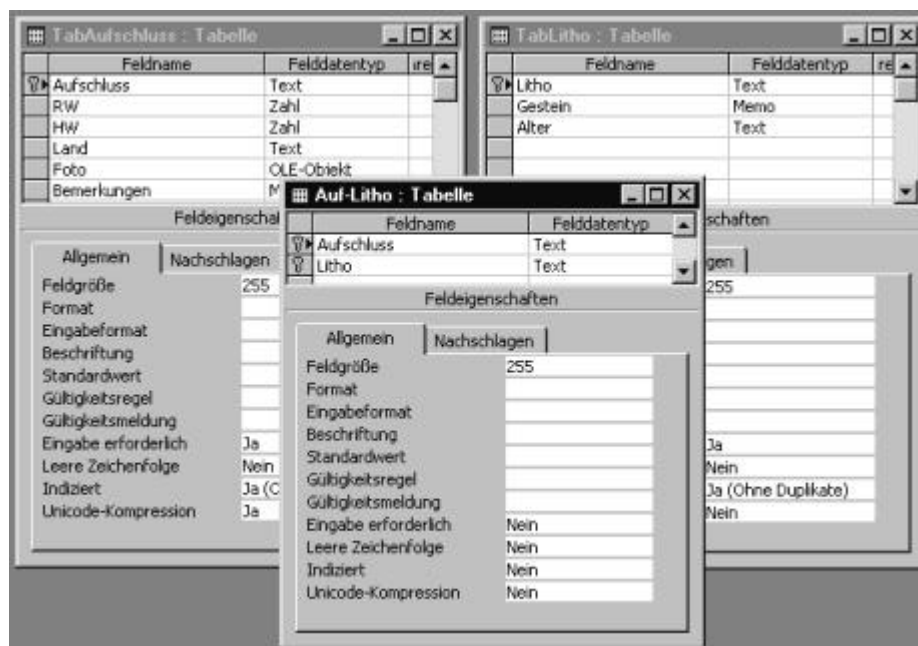
Aufschluss

Litho

Wichtig: bei beiden Feldern in dieser Tabelle muss jetzt *Eingabe erforderlich - Nein* und *Indiziert - Nein* eingetragen werden.

Die Frage nach dem *Primärschlüssel* stellt sich nun. Man könnte ein eigenes Primärschlüsselfeld namens ID einfügen; das wäre jedoch unnötige Verschwendung von Speicherplatz. Man kann **beide** Felder als Primärschlüssel definieren, da jede Kombination von *Aufschluss* und *Litho* einzigartig ist.

Dazu markiert man beide Zeilen, klickt mit der rechten Maustaste darauf und wählt den Primärschlüssel



Verknüpfen und Beziehungen definieren:

Menü *Extras - Beziehungen*. Markieren der Tabellen und *Hinzufügen*.

Nun zieht man *Aufschluss* aus *TabAufschluss* (hier unsere Tabelle "1" bzw. 'Haupttabelle') mit der Maus auf *Aufschluss* in *Auf-Litho* (hier unsere Tabelle "N", bzw. 'Nebentabelle').

Dann markiert man die drei Fenster (mit *referentieller Integrität, etc.*) und klickt *Erstellen*.

Dann macht man dasselbe mit *Litho* in *TabLitho* (Haupttabelle) und *Auf-Litho* (Nebentabelle)



Somit habe ich eine N:M Beziehung durch zwei 1:N Beziehungen mit einer zusätzlichen Tabelle erstellt.

Übung: Öffne die beiden Tabellen untereinander und gib die Daten ein

Die Dateneingabe über die Tabellen kann nun sowohl über *TabAufschluss* als auch über *TabLitho* erfolgen.

Zuerst werden wir in *TabLitho* sämtliche lithostratigraphischen Einheiten samt Lithologie und Alter eintragen. Dann öffnen wir *TabAufschluss* und fügen die Daten für den Aufschluss 'Barmstein' ein. Wenn wir das "+" Symbol anklicken, öffnet sich ein Fenster. Hier kann ich jetzt 'Barmsteinkalk' eintragen. 'Barmsteinkalk' ist der *Primärschlüssel* des entspr. Datensatzes aus *TabLitho* und somit sind die Datensätze verknüpft.

Versuche, 'Barmsteinkalk' falsch zu schreiben und Access wird dir mitteilen, dass es keinen entspr. Datensatz gibt!

Litho	Gestein	Alter
Barmsteinkalk	grobdetritische Kalkbänke	Tithonium bis Berriasium
Oberalmerschichte	bankiger mikritischer Kalk	Tithonium bis Berriasium
Plassenkalk	massiger Riffkalk	Kimmeridgium bis Berriasium
Tressensteinkalk	massige Kalkbrezlien	Kimmeridgium bis Tithonium

Aufschluss	RW	HW	Land	Foto	Bemerkung
Barmstein				or 3.0-Photo	
Barmsteinkalk					
Plassen					
Trattberg				or 3.0-Photo	
Tressenstein				or 3.0-Photo	

Der unterschied zur 1:N Verknüpfung in Beispiel 2 ist, dass wir es hier nicht mit einem Haupt- und Unterdatenblatt zu tun haben, sondern sozusagen mit zwei Hauptdatenblättern. Daher kann man bei der Erstellung eines Formulars wählen, welches der zwei Datenblätter als Unterformular dienen soll.

Übung: Geben Sie die Daten in die Tabelle ein. Erstelle nun eine Abfrage (z.B. in welchen Aufschlüssen kommt Barmsteinkalk vor).

15.1. Dateneingabe über Formulare in N:M Beziehungen

Die Eingabe über Formulare in N:M Beziehungen sind weniger einfach zu erstellen als in 1:N Beziehungen.

Dazu darf man als Unterformular nicht einfach eine andere Tabelle einfügen, sondern eine Abfrage aus Verbindungstabelle (bei uns *Auf-Litho*) und 'Unter'tabelle (bei uns *TabLitho*).

Erstellen der Abfrage: Neue Abfrage in Entwurfsansicht (d.h., nicht den Assistenten verwenden). Die Tabellen *Auf-Litho* und *TabLitho* in das obere Abfragefenster einfügen.

Aus *Auf-Litho* benötigen wir: *Aufschluss* und *Litho*

Aus *TabLitho*: *Gestein* und *Alter*.

Wir speichern die Abfrage unter ***QuelleUFmLitho*** (weil dieses Abfrage die Quelle für unser Unterformular darstellen wird).

Erstellen des Formulars:

Formular - Neu - Formular-Assistent.

- Aus *TabAufschluss* wählen wir alle Felder
- Aus *QuelleUFmLitho* wählen wir ebenfalls alle, ausgenommen *Aufschluss*

Für das Hauptformular nehmen wir *TabAufschluss*. Layout: Tabellarisch.

Namen: *FmAufschluss* für das Hauptformular und *UFmLitho* für das Unterformular

Formatieren der Formulare

Wie schon vorher, haben wir jetzt zwei Formulare.

Wir öffnen *UFmLitho* separat und passen das Layout an. Nach dem Schließen des Formulars formatieren wir *FmAufschluss*.

In *UFmLitho* verändern wir jetzt *Litho* in *Kombinationsfeld* (rechte Maustaste - Ändern - Kombinationsfeld).

Dann noch einmal mit rechter Maustaste - *Eigenschaften - Daten - Datenherkunft - TabLitho* auswählen.

Jetzt kann ich zu jedem *Aufschluss* die vorhandenen Einheiten zuweisen.

Im Gegensatz zum 1:N Formular kann ich aber keine neuen Daten in das Unterformular eingeben.

Eingabe neuer lithostratigraphischer Einheiten.

Dafür erstelle ich ein Makro zum Öffnen von *TabLitho*. Dann erstelle ich im Formular eine Befehlsschaltfläche, die dieses Makro ausführt.

Wenn ich hier in dieser Tabelle eine neue Einheit eingabe, sehe ich sie aber noch nicht im Formular! Dazu muss ich eine Befehlsschaltfläche mit dem Befehl *Formulardaten aktualisieren* einfügen.


NEU2-FmAufschluss

Aufschluss **Barmstein**

RW

LW

Land



Bemerkung

Litho	Gestein	Alter
Barmsteinkalk	Kalk	
Oberalmerschichten	Kalkstein	

Datensatz: von 3

TabLitho : Tabelle

	Litho	Gestein	Alter
+	Adneter Kalk	Cephalopodenk	
+	Barmsteinkalk	Kalk	
+	Oberalmerschichten	Kalkstein	
▶	Plassenkalk	Rifkalk	
+	Tressensteinkalk	Detrituskalk	
*			

Datensatz: von 5

16. KALKULATIONEN, AGGREGATFUNKTIONEN, DATUM

Beispiel 4 - Kostenaufstellung

Erstelle neue Datenbank und nenne diese **Kosten**

Erstelle neue Tabelle in Entwurfsansicht und füge folgende Felder ein:

1. *Kosten-ID* - definiere als Primärschlüssel, Felddatentyp: AutoWert
2. *Kaufdatum* - Felddatentyp: Datum/Uhrzeit
3. *Objekt* - Felddatentyp: Text
4. *Einzelpreis* - Felddatentyp: Zahl, Feldeigenschaften-Format: Euro
5. *Anzahl* - Felddatentyp: Zahl

Abspeichern als **TabKosten**

Kosten-ID	Kaufdatum	Objekt	Einzelpreis	Anzahl
1	02.01.01	Geologenhammer	50,00 €	1
2	04.02.01	Meissel	10,00 €	3
3	05.03.01	Geologenkompass	500,00 €	1
4	23.04.01	Geländebuch	5,00 €	2
5	30.05.01	Probensäcke	1,00 €	100
*(AutoWert)			0,00 €	0

Datensatz: 1 von 5

16.1. EINFACHE BERECHNUNGEN

Im Gegensatz zu Excel können innerhalb der Tabelle keine Kalkulationen durchgeführt werden.

Für Berechnungen muß daher eine Abfrage erstellt werden.

Erstelle eine neue Abfrage in Entwurfsansicht. Füge alle Felder (ausgenommen *Kosten-ID*) ein. In die letzte Spalte schreibe in *Feld*: **Gesamt: [Einzelpreis]*[Anzahl]**

Feld:	Kaufdatum	Objekt	Einzelpreis	Anzahl	Gesamt: [Einzelpreis]*[Anzahl]
Tabelle:	TabKosten	TabKosten	TabKosten	TabKosten	
Sortierung:					
Anzeigen:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kriterien:					
oder:					

Abspeichern unter **AbGesamtpreis**

Wechsle in die Datenblatt-Ansicht. Der Gesamtpreis (Einzelpreis x Anzahl) wurde errechnet.

Operatoren für Berechnungen:

- ^ Potenzierung
- / Division
- + Addition
- Subtraktion
- * Multiplikation

Sie können beliebig lange Formeln bilden, die z.B. auch Felder mit Zahlen multiplizieren - z.B.
Ausdruck: [Einzelpreis]*10

Suchfunktionen, z.B. für das Feld Einzelpreis:

- 50** findet alle Objekte mit einem Preis von 50 EUR
- >50** alle mit einem Preis größer als 50
- <50** kleiner als 50
- >10 UND <50** zwischen 10 und 50

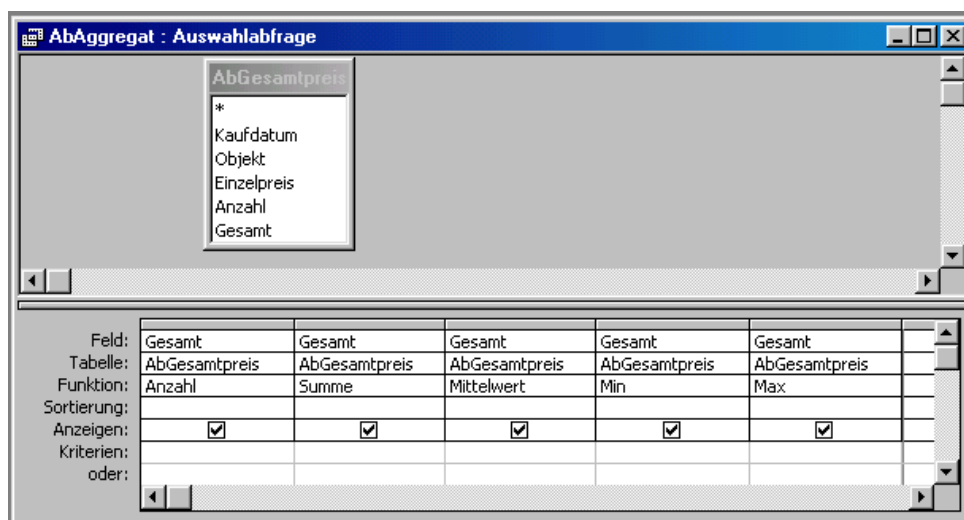
16.2. AGGREGATFUNKTIONEN

Aggregatfunktionen sind z.B. Mittelwert, Anzahl, Maximum, Minimum.

Wir können nun z.B. den Mittelwert aller Nettobeträge aus der Tabelle anhand einer neuen Abfrage errechnen.

Wir wollen aber den Mittelwert der Gesamtkosten berechnen und müssen daher eine Abfrage basierend auf der vorhergehenden Abfrage erstellen.

Dazu öffnen wir eine neue Abfrage in der Entwurfsansicht und fügen die ABFRAGE **AbGesamtpreis** ein. Daraus fügen wir das Feld Gesamt fünf mal ein und klicken im Menü auf Σ . In der Zeile Funktion wählen wir nun Anzahl, Summe, Mittelwert, Min und Max aus.



Das Ergebnis in der Datenblattansicht sieht nun so aus:

	Anzahl	Summe	Mittelwert	Min	Max
	5	690	138	10	500

Datensatz: 1 von 1

Operatoren für Aggregat-Abfragen

Summe: Summe aller Werte in einer Spalte.

Mittelwert: Durchschnittlicher Werte aller Werte einer Spalte

Minimum: Niedrigster Wert einer Spalte

Maximum: Höchster Wert einer Spalte

Anzahl: Anzahl der Einträge in einer Spalte. Nullwerte (leere Felder oder die Zahl Null) werden übergangen. Diese Funktion ist auch Texteinträge möglich

StdAbw: Standardabweichung

Varianz: Varianz = Quadrat der Standardabweichung

Erster Wert: Feldwert des ersten Datensatzes

Letzter Wert: Feldwert des letzten Datensatzes

16.3. MANIPULATION VON MIT DATUMSANGABEN

Wir öffnen wieder **AbGesamtpreis** in der Entwurfsansicht.

Wenn mich nun nur die Werte eines bestimmten Zeitraumes interessieren, kann ich dies ins Feld Kaufdatum schreiben.

Feld:	Kaufdatum	Objekt	Einzelpreis
Tabelle:	TabKosten	TabKosten	TabKosten
Sortierung:			
Anzeigen:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Kriterien:	>#01.03.01#		
oder:			

Z.B. wird der Eintrag >01.03.01 alle Werte nach diesem Datum anzeigen (für Uhrzeiten gilt das selbe).

Operatoren für Datums-Abfragen

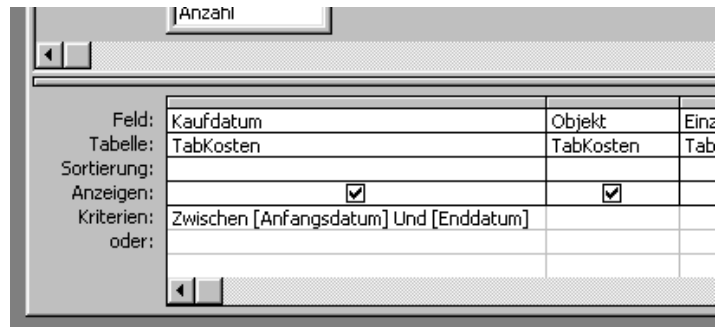
>#31.12.99#: alle Einträge nach diesem Datum

<#31.12.99#: alle Einträge vor diesem Datum

Zwischen #31.12.99# Und #31.12.00#: alle Einträge zwischen diesen Daten, einschließlich 31.12.99 und 31.12.00

Für Uhrzeiten gelten die selben Operatoren!

Diese Abfrage kann auch als Parameter-Abfrage ausgeführt werden. Der Eintrag **zwischen** [Anfangsdatum] Und [Enddatum] fragt bei der Ausführung der Abfrage nach dem Zeitraum, der angezeigt werden soll. Die Worte zwischen den eckigen Klammern sind beliebig wählbar.



	Objekt	Einzel
Feld:	Kaufdatum	
Tabelle:	TabKosten	Tab
Sortierung:		
Anzeigen:	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Kriterien:	Zwischen [Anfangsdatum] Und [Enddatum]	
oder:		

Die Aggregatabfrage **AbAggregat** basiert ja auf der Abfrage **AbGesamtkosten**. Wenn ich nun **AbAggregat** aufrufe, werde ich nach dem Zeitraum gefragt, für den Mittelwert etc. angezeigt werden soll.

DATENBANKPFLEGE

Achtung Datensicherheit:

Das Löschen von Datensätzen kann nicht mehr rückgängig gemacht werden

Versehentliche Änderungen an Daten in Tabellen, Abfrageergebnissen und nicht gesperrten Formularen werden sofort ausgeführt. D.h., Access fragt nicht, ob die Änderungen übernommen werden sollen.

Zur Absicherung kommen zwei Möglichkeiten in Frage: (1) Das Ansehen und Durchblättern vorhandener Daten nur in Berichten durchführen, dort können Daten nicht verändert werden; oder (2) ein Formular für die Dateneingabe und ein zweites für das Durchblättern der Daten; im zweiten Formular können dann beliebige Felder gesperrt werden (Über die *Eigenschaften - Daten - Gesperrt: Ja*)

Zur **Datenbankpflege** zählen Löschen und Aktualisieren sowie Komprimierungen. Sie dient dazu, Datenbanken klein zu halten und ihre Leistung und Sicherheit zu optimieren.

Die **Duplikatsuche** ermöglicht das Löschen von doppelt vorhandenen Datensätzen. Z.B. kann ich versehentlich zwei Einträge zu einer Lokalität mit unterschiedlicher ID haben (dem kann allerdings vorgebeugt werden, indem die Lokalität als Primärschlüssel definiert wird).

Vorgangsweise: neue Abfrage mit Assistenten erstellen und Assistent zur Duplikatsuche wählen.

Die **Inkonsistenzsuche** ermöglicht das Aufsuchen von Datensätzen, die nicht miteinander verknüpft sind und dadurch u.U. Karteileichen darstellen (z.B. eine Probe die keiner Lokalität zugeordnet ist).

Vorgangsweise: neue Abfrage mit Assistenten erstellen und Assistent zur Inkonsistenzsuche wählen.

Bei Absturz des PC kann die Db. beschädigt werden. In diesem Fall wählt man im Menü

Extras -> Datenbank-Dienstprogramme -> **Datenbank komprimieren und reparieren.**

Beim Löschen von Tabellen, Abfragen, Formularen, etc., werden diese NICHT wirklich aus der Datei gelöscht. Dies führt zu unnötig großen Dateien. Abhilfe schafft man mit einer **Komprimierung**:

Extras -> Datenbank-Dienstprogramme -> Datenbank komprimieren und reparieren.

Die Komprimierung kann entweder an einer geöffneten oder einer geschlossenen Db. durchgeführt werden.

Die **Datenbankaufteilung** erlaubt eine Aufgliederung in zwei Dateien. Dies geschieht im Menü:

Extras -> Datenbank-Dienstprogramme -> Assistent zur Datenbankaufteilung.

Dabei wird eine Datenbank mit den Tabellen (=Daten) und eine mit den Abfragen, Formularen, etc. erstellt.

Großer Vorteil: ist die Anwendungsdatei kaputt, sind die Daten immer noch vorhanden. Dies ist bei einer einzigen Datei nicht gesichert.

MDE-Dateien erstellen. Damit wird eine Kopie der Originaldatenbank (z.B. Proben.mdb wird zu Proben.mde) erstellt, die den Quellcode nicht enthält. D.h., der Anwender kann keine Befehlszeilen mehr verändern und nichts ruinieren. Außerdem wird die Datenbank dadurch kleiner und leistungsoptimiert.