**Start:**

Tabelle löschen,

Start => Optionen => Aktuelle Datenbank

\*Anwendungstitel

\*Entwurfsänderung für Tabellen deaktivieren, damit keine Felder von Benutzern hinzugefügt werden können

Datei schließen und öffnen.

=> Tabellenentwurf anlegen

**Datentypen:**

**WICHTIG => Speicher sparen => Immer richtigen Datentyp und Größe eingeben!!!**

Für Telefonnummern mit /, Leereichen, + usw… => kurzer Text, Feldgrö0e 50

Währung => Format Euro, ev. Standardwert auf NULL stellen (löschen)

Wenn nur ganzzahlige 0-9, dann Byte

**Feldnamen:**

Keine neutralen Namen wie Name, Datum, usw. verwenden, besser Lieferdatum

Bei Datentypen gibt es eine automatische ENDE Markierung, es werden also keine Leerzeichen angehängt

Berechnete Felder funktionieren nur wenn Tabelle gespeichert ist!

Zum Beispiel:

[adresNameNach] & ", " & [adresNameVor]

Feld&„Komma Leerzeichen“&Feld

& " in " & [adresOrt]

Gültigkeitsregel!

Zum Beispiel Text ist mindestens 4 Chars lang:

=> Ausdrucks-Generator => Funtkionen => iFunktionen => Länge([adresPLZ] ?Feld)>=4

Wichtig:

Länge(„adresPLZ“) misst wie lange adresPLZ ist, also den String.

Länge([adresPLZ]) misst, weil lange der Wert im Feld ist!

Gültigkeitsmeldung => Netter und schöner machen!

Auf keinen Fall mehrere Inhalte in ein Feld schreiben! (2 Telefonnummern, 2 Bestellungen usw.… in ein Feld)

**Normalformen de relationalen Datenbanken**

**1.Normalform: Unteilbarkeit (Atomisierung)**

=> Mehrer Inhalte in einem Feld sind nicht erlaubt

=> Zusammengesetzte Inhalte sind unbrauchbar, also:

Wenn mehrere Inhalte da, in eigene Datenbanken mit 1(n) auslagern und nicht mehrere Spalten der in eine Spalte schreiben

**2.Normalform: Redundanz**

=> Daten, die von einem Feld des gleichen Datensatzes abhängig sind, dürfen nicht in der gleichen Tabelle gespeichert werden!

Also zum Beispiel: Nachschlagelisten ZWINGEND ! oder Bruttobeträge zwingend berechnen! Werte, die errechnet werden können, errechnen. Werte, die nachgeschlagen werden können, nachschlagen.

**3. Normalform: Historie**

=>Daten, die direkt von einem anderen Wert abhängig sind, dürfen nicht in der gleichen Tabelle gespeichert werden!

Zum Beispiel: Preise, die sich verändern nicht direkt in der Datenbank ändern, sondern eigene „historische Preise“ Datenbank erstellen, in der alle Preisänderungen eingetragen werden. Von dort bezieht dann die Haupt-Datenbank die aktuell gültigen Preise.

**3 Phasen der Aufteilung der Daten nach bestimmten Regeln:**

**Storming**

Unkontrolliertes sammeln der nötigen Daten

**Norming**

Einteilen der Daten in für die Datenbank sinnvolle Kategorien und Abhängigkeiten

**Performing**

Beschleunigung der Performance