

**„FBConsole“
mit der Datenbank
„Firebird“**

**Dipl. Inf., Dipl.-Ing. (FH) Michael Wilhelm
Friedrichstraße 57 - 59
38855 Wernigerode**

**Raum: 2.202
Tel.: 03943/659-338
Fax: 03943/659-399
Email: mwilhelm@hs-harz.de
Web: <http://www.miwilhelm.de>
Link: <http://www.miwilhelm.de/scripte/download/dbs/Datenbank.zip>**

Inhaltsverzeichnis

1	Zip-Datei	4
1.1	Installation	4
1.2	Inhalt der ZIP-Datei	4
2	Datenbank-Designer.....	5
1.1	Eigenschaften des Programms	5
1.2	Die Eigenschaften in Kurzform:	5
1.3	Ablauf bei der Erstellung eines logischen Modells	6
2.1	Beispiel eines logischen Modells	7
2.2	Erstellen der SQL-Skripte	18
2.3	Ergebnis:	19
3	FBConsole.....	21
3.1	Neue Datenbank erstellen	22
3.2	Daten eintragen	24
3.2.1	SQL-Befehle zum Eintragen:	24
3.3	Testabfragen	25
4	Anhang	27
4.1	Generator	27
4.2	Sequenz	27
4.3	Daten-Eintrag	28
5	Indexverzeichnis	29

Abbildungsverzeichnis

Abbildung 1 Überblick des Programms	5
Abbildung 2 Auswahl der Datenbank	7
Abbildung 3 Neues Entity "Artikel"	7
Abbildung 4 PK-Constraint „PK_Artikel“ einfügen	7
Abbildung 5 Primary-Key eintragen	9
Abbildung 6 Bezeichnung einfügen	9
Abbildung 7 Default-Wert eintragen (ohne Hochkommas)	10
Abbildung 8 Check-Bedingung	10
Abbildung 9 Attribut Gewicht	11
Abbildung 10 Defaultwert	11
Abbildung 11 Check-Bedingung	12
Abbildung 12 Vollständiges Entity Artikel	12
Abbildung 13 Darstellung im Hauptfenster	12
Abbildung 14 Entity Kunde	13
Abbildung 15 Entity Bestellung	13
Abbildung 16 Relation definieren	14
Abbildung 17 Hauptfenster der Relationen	14
Abbildung 18 Frage nach dem Eintragen	14
Abbildung 19 Kundennr wird eingefügt	15
Abbildung 20 Hauptfenster mit der ersten Relation	15
Abbildung 21 Zweite Relation	16
Abbildung 22 Zweite Relation erstellt	16
Abbildung 23 Hauptfenster mit der ersten und zweiten Relation	17
Abbildung 24 Erzeugen der SQL-Skripte	18
Abbildung 25 Liste der Tabellen	19
Abbildung 26 Hauptfenster der FBConsole	21
Abbildung 27 Erstellen der Datenbank "Firma.fdb"	22
Abbildung 28 Eintragen der SQL-Skripte	23
Abbildung 29 Anzeige der Datenbank-Tabellen	23
Abbildung 30 Daten mittels „Insert-Into“ eintragen	24
Abbildung 31 Abfrage bezüglich der Artikel	25
Abbildung 32 Abfrage bezüglich der Kunden	25
Abbildung 33 Abfrage bezüglich der Bestellungen	26
Abbildung 34 Abfrage bezüglich der Bestellungen	26
Abbildung 35 Definition eines Generators	27

1 Zip-Datei

Dieses Script bietet einen Einstieg in die Benutzung von Datenbanken unter Winform / WPF. Dabei wird ein Gesamtpaket vom Datenbank-Designer, über eine Datenbank-Console bis hin zu grafischen Programme bereitgestellt.

1.1 Installation

- Entzippen der Datei in einen beliebigen Ordner

1.2 Inhalt der ZIP-Datei

Ordner Designer

- Datenbank-Designer
 - Programm „DBW_Designer.exe“
 - Datenbank-Designer-Handbuch.pdf
 - Ordner „dbs“
 - Ordner „def“

Ordner Firebird

- Datenbank_Employee.pdf
- dbs_Firebird_Oracle.pdf
- Tabelle der Firebird-Beispieldatenbank
 - Employee.fdb
 - Oracle.fdb
 - Eisenbahn.fdb
- Firebird-Datenbank-Konsole, nur Programm
- DLL's
 - fbembed.dll
 - FirebirdSql.Data.FirebirdClient.dll
 - icudt30.dll
 - icuin30.dll
 - icuuc30.dll

2 Datenbank-Designer

Das vorliegende Programm soll den Datenbankentwurf vereinfachen. Dazu werden die Elemente (Entities, Relationen etc), grafisch dargestellt. Mit einfachen Befehlen können die Elemente erzeugt und bearbeitet werden.

Es wird vorausgesetzt, dass Grundbegriffe in der Datenbanktechnologie vorhanden sind.

1.1 Eigenschaften des Programms

Die Abbildung 1 zeigt den Aufbau des Designers. Es hat die normale Menü- und Schalterstruktur. Alle Datenbankelemente werden in einem Baum links dargestellt. Die Breite dieses Baumes kann mit der blauen Linie (Splitter) verändert werden. Im rechten Hauptteil werden die Entities und Beziehungen grafisch dargestellt.

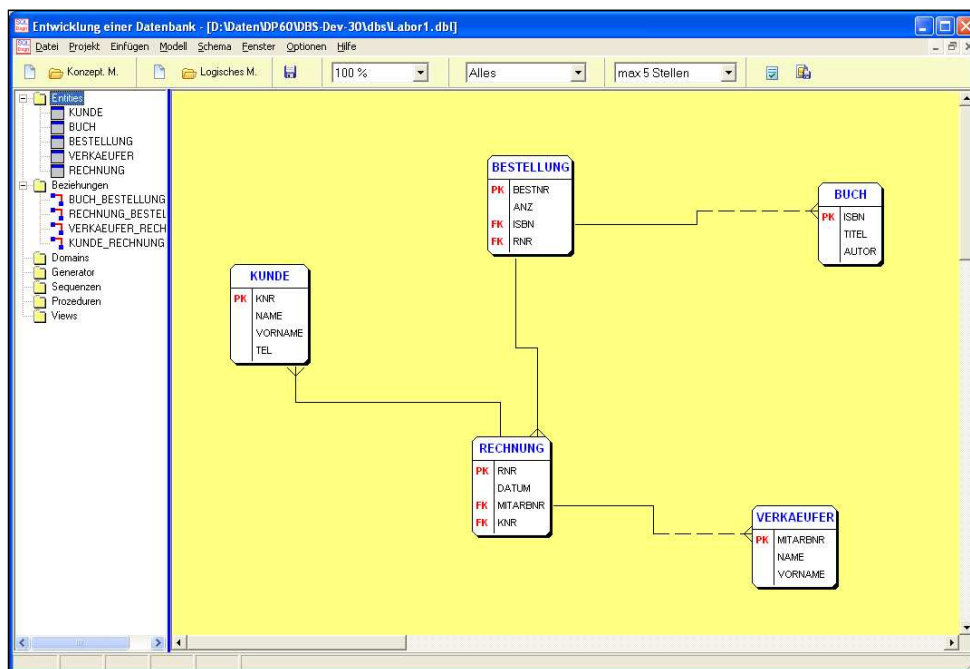


Abbildung 1 Überblick des Programms

1.2 Die Eigenschaften in Kurzform:

- Verwendung des konzeptionelles Modell
- Verwendung des logischen Modell
- Automatischer Transfer vom konzeptionellen zum logischen Modell
- Definition der Datenbanktypen (logische Modell)
- Vordefinierte Datenbanken
- **Freie Definition einer Datenbank**
 - Namen der Datentypen
 - Zuordnung zu den Grunddatentyp

- SQL-Syntax definieren (Kommentare, Eckige Klammer etc.)
- Counter definieren (Generator, Sequenz, Autonumber)
- Liste der reservierten Wörter
- Erzeugung und Verwaltung von Entities
- Erzeugung und Verwaltung von Attributen
- Erzeugung und Verwaltung von Relationen
- Erzeugung und Verwaltung von Generatoren
- Erzeugung und Verwaltung von Sequenzen
- Generierung eines SQL-Scriptes
- Vielfältige Farb- und Schriftwahl
- Anzeige der Relationen mit Krähenfuss oder Punktnotation
- Export der Daten nach Winword
- Export der ERM-Grafik in die Zwischenablage
- Es kann eine zusätzliche Instanz aufgerufen werden (ALT-Tab Umschaltung)
- Kopieren von Entities über die Zwischenablage möglich
- Export in die Zwischenablage als Bild mit Randdefinition
- Überprüfung der Datenbank auf doppelte Namen, Checkbedingungen
- Check-Attribut in Entity-Einst eingetragen

Konzeptionellen Modell:

- Es kann nun auch ein Defaultwert eingegeben werden
- Es kann nun auch eine Check-Bedingung eingegeben werden
- Die Länge der Kurznamen wird auf 8 Zeichen begrenzt
- Es gibt einen Test der Relationsnamen FK_???? auf Länge <=31 Zeichen
- Es gibt nun auch eine Weak-Relation
- Die Anzeige erfolgt in der Chen-Notation und Martin-Notation
- Es gibt Multi-Attribute in den Entities
- Es gibt Ternäre Relationen
- Es gibt auch eine Self-Relation
- Rauten-Darstellung der Relation
- Die Relationen können Attribute haben

Logische Modell:

- Es gibt nun auch ein Weak-Entity
- Die Anzeige erfolgt nun auch in der Martin-Notation

1.3 Ablauf bei der Erstellung eines logischen Modells

- **Erzeugen eines neuen Fensters (STRG+N)**, nur wenn es kein konzeptionelles Modell gab.
 - Datenbank: Firebird, Version 2,5
- Einstellen der Projekteinstellungen (Darstellungen der Beziehungen, Texte)
- **Erstellen der Entities (STRG+E)**
- **Eintragen der Attribute, ohne der/des Fremdschlüssels**
- Festlegen der Primärschlüssel
- **Erstellen der Beziehungen (STRG+R)**
- **Zusätzliche Eintragungen (Check, Unique, Weak-Entities, Generator, Sequenz)**
- Eintragen der Texte (Beschreibungen etc.)
- **Aufruf der Generierung zum SQL-Skript (F9)**

2.1 Beispiel eines logischen Modells

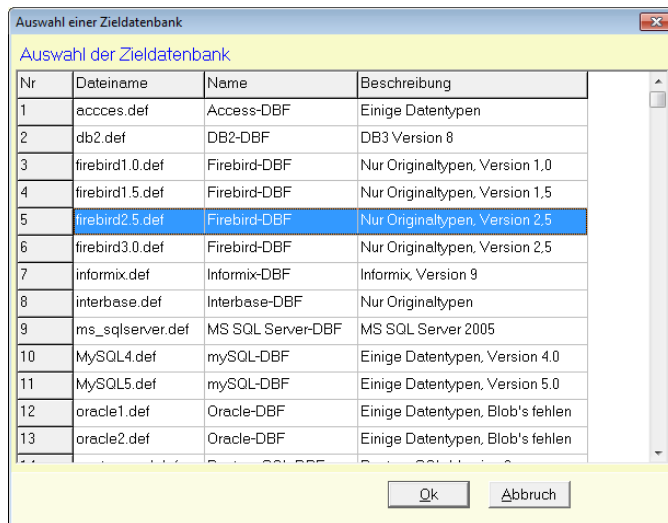


Abbildung 2 Auswahl der Datenbank

Hinweis:

Bitte wählen Sie für dieses Programmpaket immer die Version „Firebird 2,5“ aus.

Neues Entity „Artikel“ mit Strg+E

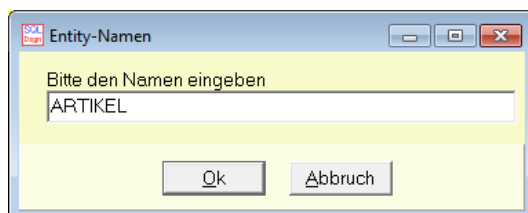


Abbildung 3 Neues Entity "Artikel"

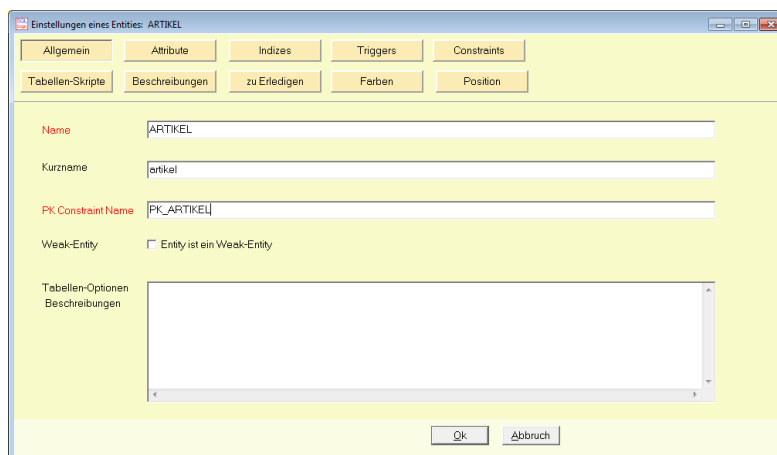


Abbildung 4 PK-Constraint „PK_Artikel“ einfügen

Attribute einfügen:

Einstellung eines Attributes: PINDEX

Allgemein Default/Unique Checks Linked-Attribute Beschreibung Zu Erledigen Object-ID

Name: ARTINDEX

☒ Primärschlüssel ☒ Not Null ☐ Fremdschlüssel

Datentyp-Definition

Domain: keine Domain Neue Domain

Datentyp: INTEGER 0 0

Formel:

Datentyp: kein Datentyp 1 0

Autonumber

☐ aktiviert Start: 0 Increment: 1

Ok Abbruch

Abbildung 5 Primary-Key eintragen

Einstellung eines Attributes: Attribut2

Allgemein Default/Unique Checks Linked-Attribute Beschreibung Zu Erledigen Object-ID

Name: BEZ

☐ Primärschlüssel ☐ Not Null ☐ Fremdschlüssel

Datentyp-Definition

Domain: keine Domain Neue Domain

Datentyp: VARCHAR 50 0

Formel:

Datentyp: kein Datentyp 1 0

Autonumber

☐ aktiviert Start: 0 Increment: 1

Ok Abbruch

Abbildung 6 Bezeichnung einfügen

The screenshot shows the 'Einstellung eines Attributs: Attribut2' dialog box with the 'Default/Unique' tab selected. The 'DefaultWert: (VARCHAR)' section has the 'Defaultwert' checkbox checked, and the 'Wert' field is empty. Below this, a red note states: 'Datum, Zeit, Boolean, Currency werden als String gespeichert (Ohne Prüfung)'. The 'Unique-Bedingungen' section has the 'Unique' checkbox unchecked, and the 'Check-C. Name' field is empty. At the bottom are 'Ok' and 'Abbruch' buttons.

Abbildung 7 Default-Wert eintragen (ohne Hochkommas)

The screenshot shows the 'Einstellung eines Attributs: Attribut2' dialog box with the 'Checks' tab selected. The 'Check-Bedingungen' section has the 'Check Attribut' checkbox checked. The 'Check-C. Name' field contains 'CH_BEZ'. The 'Check-Bedingung' text area contains the SQL code:

```
CHECK(  
  BEZ <> ''  
)
```

 At the bottom are 'Ok' and 'Abbruch' buttons.

Abbildung 8 Check-Bedingung

Einstellung eines Attributes: Attribut3

Allgemein | Default / Unique | Checks | Linked-Attribute | Beschreibung | Zu Erledigen | Object-ID

Name: GEWICHT

☐ Primärschlüssel ☐ Not Null ☐ Fremdschlüssel

Datentyp-Definition

Domain: keine Domain Neue Domain

Datentyp: DOUBLE PRECISION 1 0

Formel:

Datentyp: kein Datentyp 1 0

Autonumber

☐ aktiviert Start: 0 Increment: 1

Ok Abbruch

Abbildung 9 Attribut Gewicht

Einstellung eines Attributes: Attribut3

Allgemein | Default / Unique | Checks | Linked-Attribute | Beschreibung | Zu Erledigen | Object-ID

DefaultWert: (DOUBLE PRECISION)

☒ Defaultwert

Wert: 0

Datum, Zeit, Boolean, Currency werden als String gespeichert (Ohne Prüfung).

Unique-Bedingungen

☐ Unique

Check-C. Name:

Ok Abbruch

Abbildung 10 Defaultwert

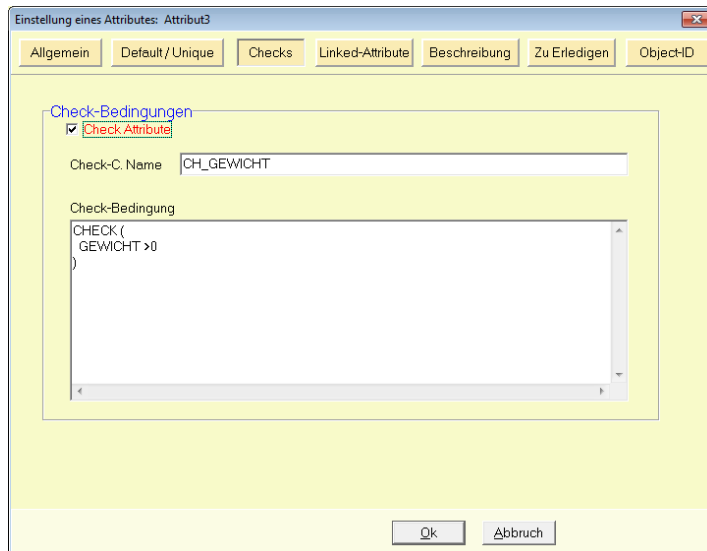


Abbildung 11 Check-Bedingung

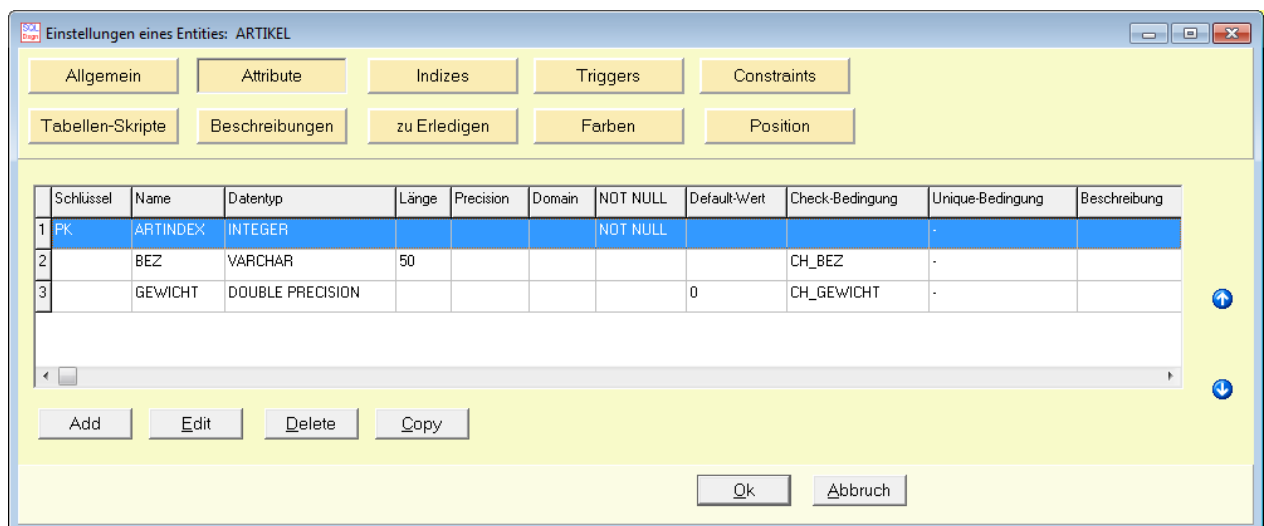
Ergebnis:

Abbildung 12 Vollständiges Entity Artikel

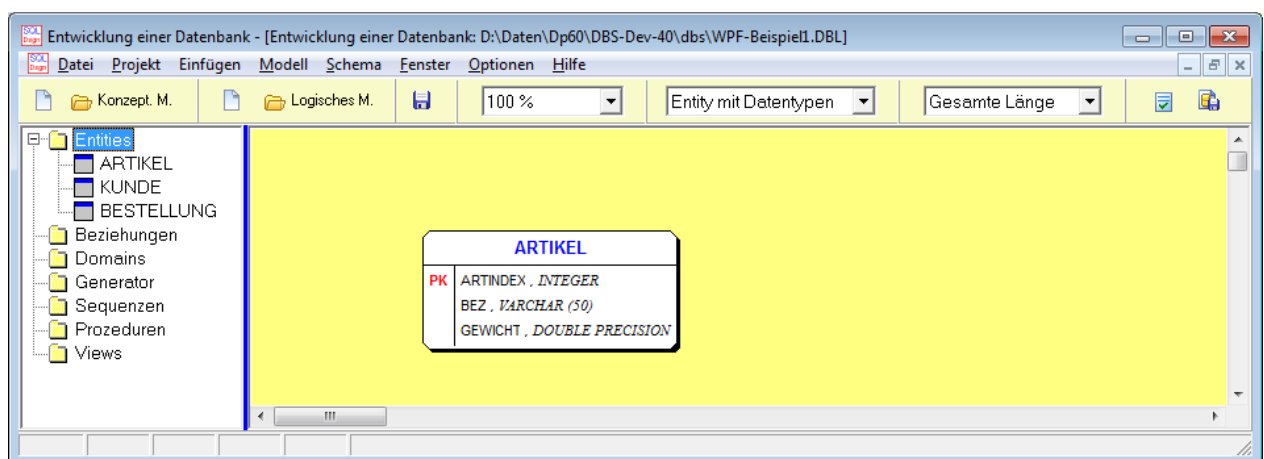


Abbildung 13 Darstellung im Hauptfenster

Entity Kunde:

Einstellungen eines Entities: KUNDE

Attribute

	Schlüssel	Name	Datentyp	Länge	Precision	Domain	NOT NULL	Default-Wert	Check-Bedingung	Unique-Bedingung	Beschreibung
1	PK	KUNDENNR	INTEGER				NOT NULL	-		-	
2		NAME	CHAR	50				-		-	
3		VORNAME	CHAR	50				-		-	
4		STRASSE	CHAR	50				-		-	
5		PLZ	CHAR	5			NOT NULL	-		-	
6		ORT	CHAR	50				-		-	

Add Edit Delete Copy

Ok Abbruch

Abbildung 14 Entity Kunde

Name hat den Defaultwert "
 Vorname hat den Defaultwert "
 Strasse hat den Defaultwert "
 Ort hat den Defaultwert "

Entity Bestellung:

Einstellungen eines Entities: BESTELLUNG

Attribute

	Schlüssel	Name	Datentyp	Länge	Precision	Domain	NOT NULL	Default-Wert	Check-Bedingung	Unique-Bedingung	Beschreibung
1	PK	BESTNR	INTEGER				NOT NULL	-		-	
2		ANZ	INTEGER					0	CH_ANZ	-	

Add Edit Delete Copy

Ok Abbruch

Abbildung 15 Entity Bestellung

Anz hat den Defaultwert null
 Anz hat die Check-Bedingung $anz > 0$

Relationen:

1. Beziehung: Kunde zu Bestellung

Taste: Strg+R



Abbildung 16 Relation definieren

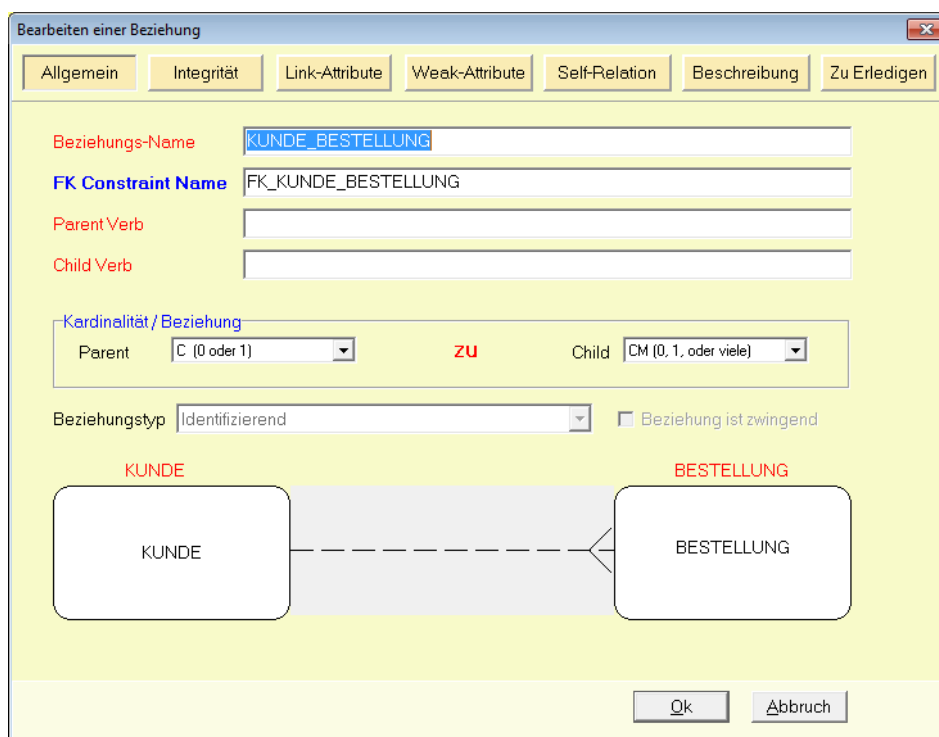


Abbildung 17 Hauptfenster der Relationen

Nun umschalten zum Register „Link-Attribute“

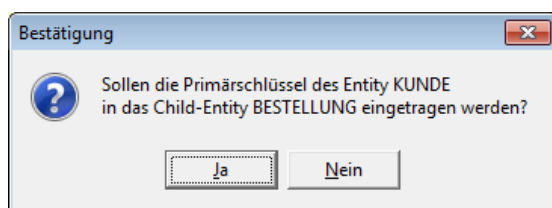


Abbildung 18 Frage nach dem Eintragen

Wenn die Bezeichnung des Primary-Key nicht in der zweiten Tabelle vorkommt, kann man die Frage bejahen. Andernfalls muss man VORHER eine zusätzliche Spalte einfügen. Also muss man dann dieses Dialogfenster beenden.

Abbildung 19 Kundennr wird eingefügt

Ergebnis:

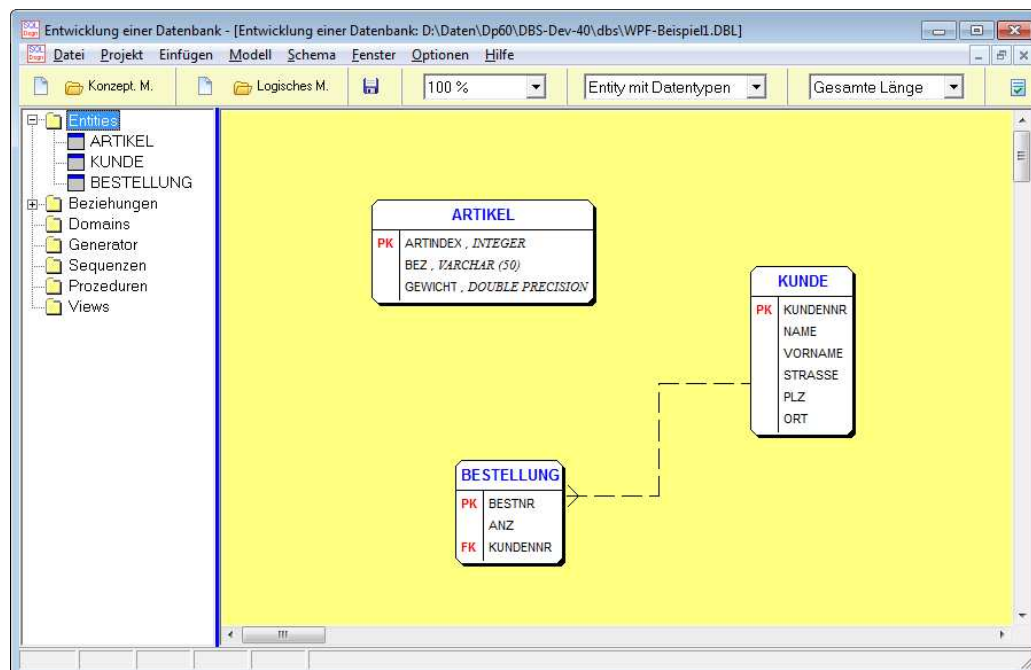


Abbildung 20 Hauptfenster mit der ersten Relation

2. Beziehung: Artikel zu Bestellung

Es kann pro Eintrag nur ein Artikel bestellt werden, aber mit beliebiger Anzahl!

Neue Beziehung

Beziehung definieren

Parent Entity (c, 1, cm, m), PK

Child Entity (c, 1, cm, m), FK

ARTIKEL

BESTELLUNG

Ok Abbruch

Abbildung 21 Zweite Relation

Bearbeiten einer Beziehung

Allgemein Integrität Link-Attribute Weak-Attribute Self-Relation Beschreibung Zu Erledigen

DisConnect

Verknüpfte Attribute

Nr	Parent-Key	Child-Key	Parent-Ind	Child-Ind	Parent-ID	Child-ID
1	ARTINDEX	ARTINDEX	0	3	5	122

Connect

ARTIKEL

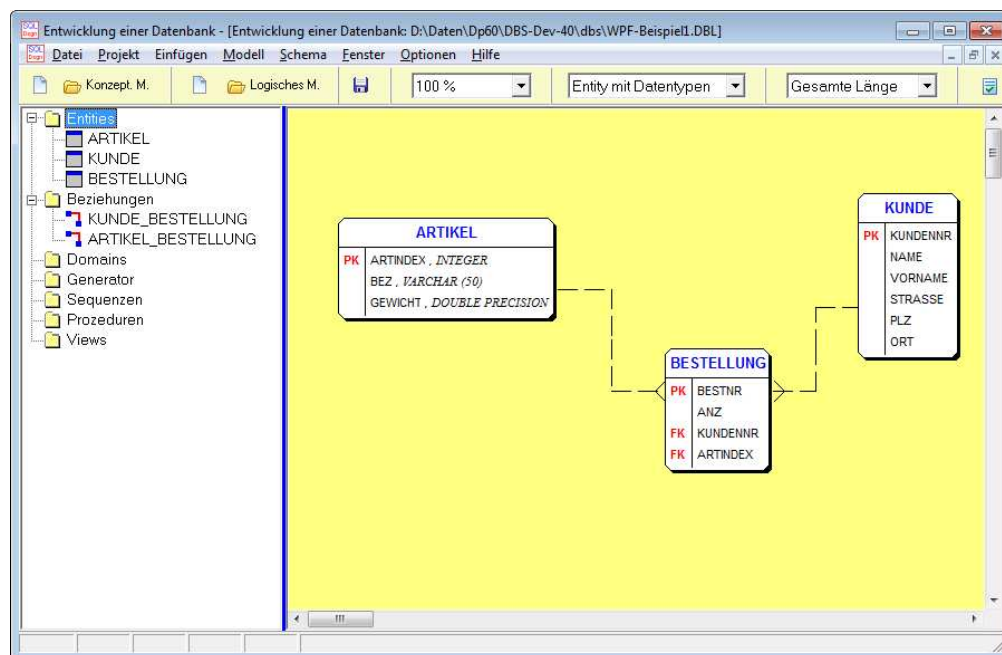
Nr	PK	FK	Attribut	Datentyp
1	PK		ARTINDEX	INTEGER
			BEZ	VARCHAR
			GEWICHT	DOUBLE P

BESTELLUNG

Nr	PK	FK	Neu	Attribut	Datentyp
	PK			BESTNR	INTEGER
				ANZ	INTEGER
		FK		KUNDENNR	INTEGER
1		FK	Neu	ARTINDEX	INTEGER

Ok Abbruch

Abbildung 22 Zweite Relation erstellt

Ergebnis:**Abbildung 23 Hauptfenster mit der ersten und zweiten Relation**

Die Datenbank ist nun fertig und die Scripte können erstellt werden.

2.2 Erstellen der SQL-Skripte

Mit dem Menü „Schema“ und dem Eintrag „Generierung Datenbank“ wird die Erzeugung des SQL-Skriptes angestoßen. Als Kurztaste existiert die Taste F9. Es erscheint folgendes Fenster (siehe untere Abbildung 24). In der obersten Zeile wird der Dateiname eingetragen. Der nächste Teil definiert die Objekte, die generiert werden sollen. Im letzten Teil sind zusätzliche Optionen vorhanden.



Abbildung 24 Erzeugen der SQL-Skripte

Im zweiten Register kann man nun einzelne Tabellen ausschließen.

Mit dem Schalter „Ok“ werden die SQL-Anweisungen erstellt.



Abbildung 25 Liste der Tabellen

2.3 Ergebnis:

```
CREATE TABLE ARTIKEL (  
    ARTINDEX INTEGER NOT NULL,  
    BEZ VARCHAR(50) DEFAULT '',  
    GEWICHT DOUBLE PRECISION DEFAULT 0,  
  
    CONSTRAINT PK_ARTIKEL PRIMARY KEY (ARTINDEX)  
);  
  
CREATE TABLE KUNDE (  
    KUNDENNR INTEGER NOT NULL,  
    NAME CHAR(50) DEFAULT '',  
    VORNAME CHAR(50) DEFAULT '',  
    STRASSE CHAR(50) DEFAULT '',  
    PLZ CHAR(5) NOT NULL,  
    ORT CHAR(50) DEFAULT '',  
  
    CONSTRAINT PK_KUNDENNR PRIMARY KEY (KUNDENNR)  
);
```

```
CREATE TABLE BESTELLUNG (  
    BESTNR INTEGER NOT NULL,  
    ANZ INTEGER DEFAULT 0,  
    KUNDENNR INTEGER NOT NULL,  
    ARTINDEX INTEGER NOT NULL,  
  
    CONSTRAINT PK_BESTNR PRIMARY KEY (BESTNR)  
);
```

```
ALTER TABLE BESTELLUNG ADD CONSTRAINT  
    FK_KUNDE_BESTELLUNG FOREIGN KEY (KUNDENNR) REFERENCES  
KUNDE(KUNDENNR);
```

```
ALTER TABLE BESTELLUNG ADD CONSTRAINT  
    FK_ARTIKEL_BESTELLUNG FOREIGN KEY (ARTINDEX) REFERENCES  
ARTIKEL(ARTINDEX);
```

```
ALTER TABLE ARTIKEL  
    ADD CONSTRAINT CH_BEZ CHECK (  
    BEZ <> ''  
);
```

```
ALTER TABLE ARTIKEL  
    ADD CONSTRAINT CH_GEWICHT CHECK (  
    GEWICHT >0  
);
```

```
ALTER TABLE BESTELLUNG  
    ADD CONSTRAINT CH_ANZ CHECK (  
    ANZ >0  
);
```

3 FBConsole

Dieses Kapitel beschreibt das Erstellen, Füllen und Abfragen einer neuen Datenbank

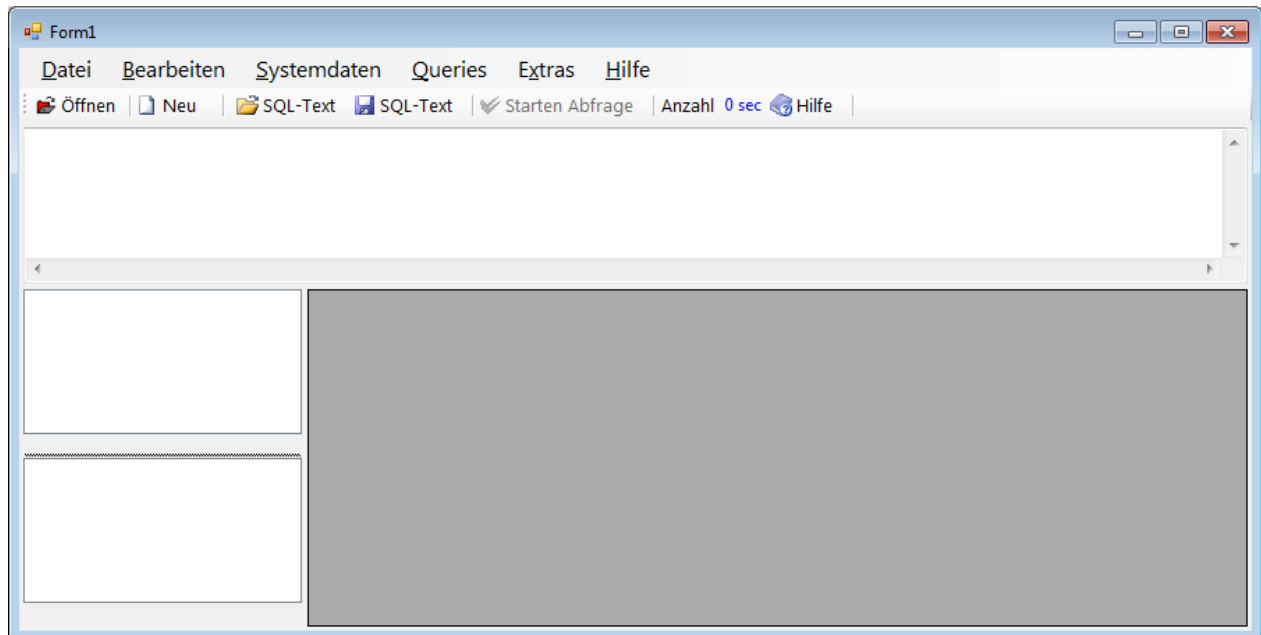


Abbildung 26 Hauptfenster der FBConsole

Schalter:

- | | |
|-------------------|---|
| • Öffnen | Öffnen einer vorhandenen Datenbank. |
| • Neu | Erstellen einer neuen Datenbank. |
| • SQL-Text | Öffnen einer vorhandenen Textdatei mit einer SQL-Anweisung. |
| • SQL-Text | Speichern der aktuellen SQL-Anweisung in eine Textdatei. |
| • Starten Abfrage | Ausführen der Anweisung im Texteditor |

Menü Datei:

- | | |
|-------------------------------|---|
| • Öffnen einer Datenbank | Öffnen einer vorhandenen Datenbank. |
| • Neue Datenbank | Erstellen einer neuen Datenbank. |
| • Öffnen einer SQL-Datei | Öffnen einer vorhandenen Textdatei mit einer SQL-Anweisung. |
| • Speichern eines SQL-Befehls | Speichern der aktuellen SQL-Anweisung in eine Textdatei. |

Menü Queries:

- | | |
|----------------------------|--|
| • Ausführen der Abfrage | Ausführen der Anweisung im Texteditor. |
| • Export der Ergebnisse | Speichern des Ergebnisses der Abfrage in die Zwischenablage. |
| • Speichern der Ergebnisse | Speichern des Ergebnisses der Abfrage in eine Textdatei. |
| • | |

3.1 Neue Datenbank erstellen

Ablauf:

- Erstellen der Skripte mittels eines Datenbank-Designers
- Erstellen einer neuen Datenbank: STRG+N

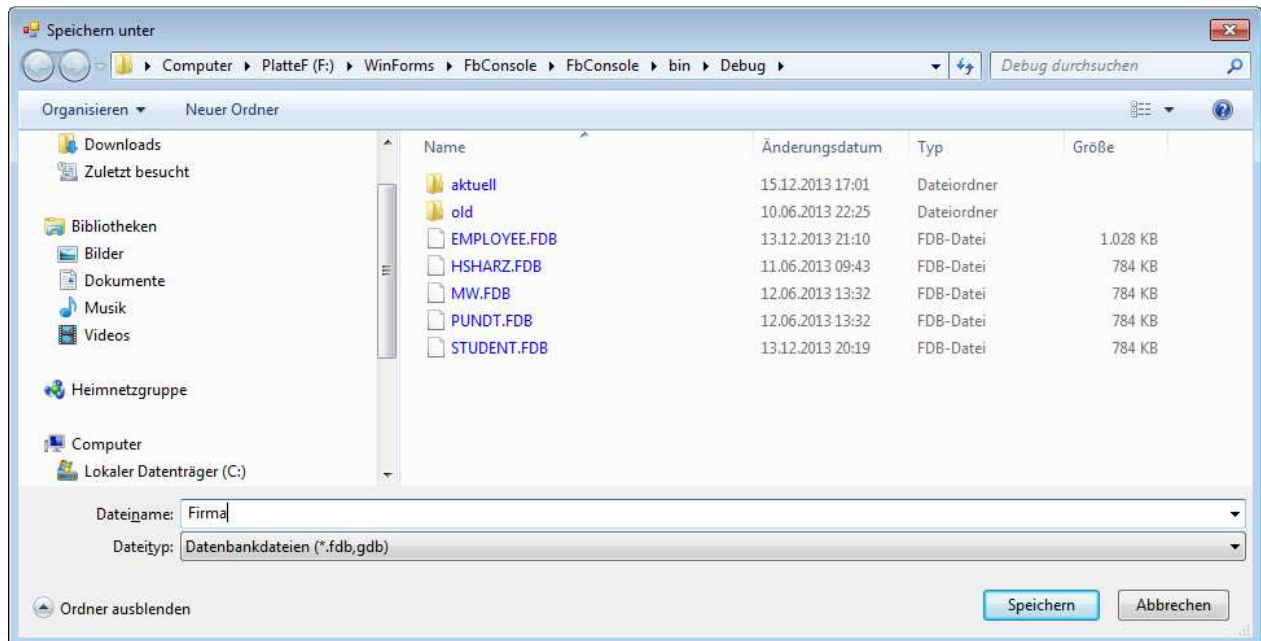
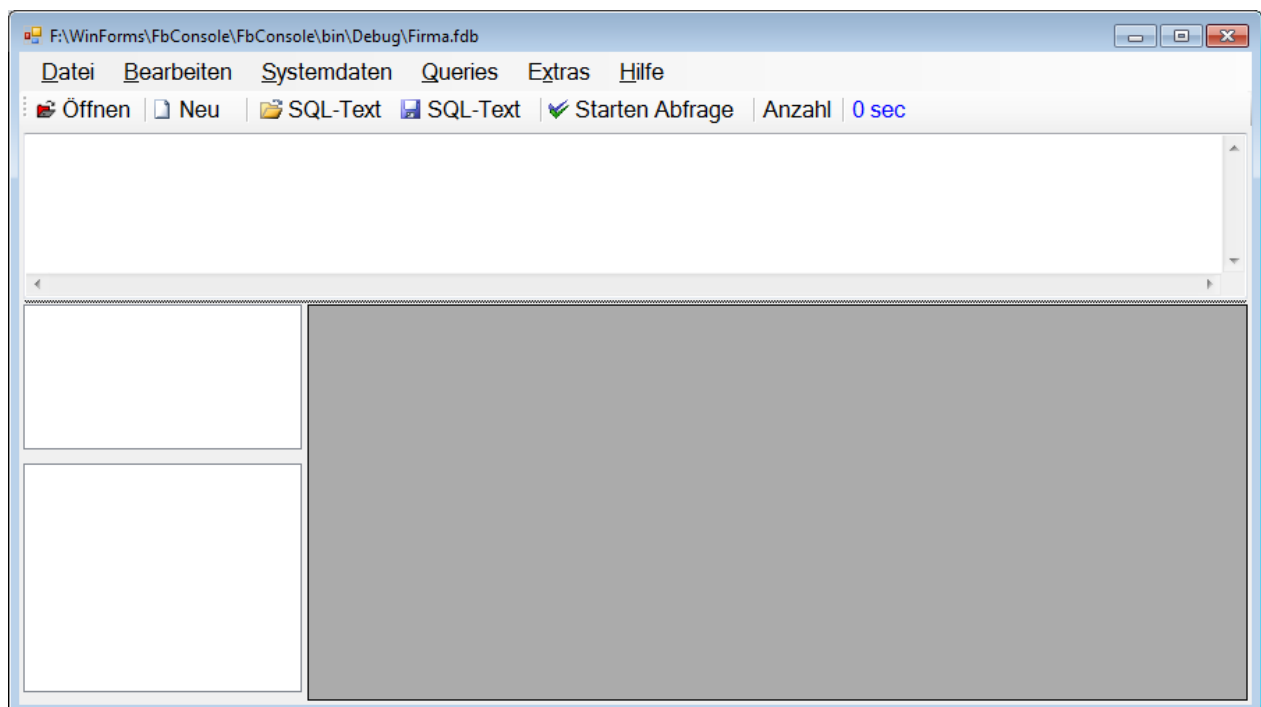


Abbildung 27 Erstellen der Datenbank "Firma.fdb"

Ergebnis:



Einfügen der Skripte in den Editor:

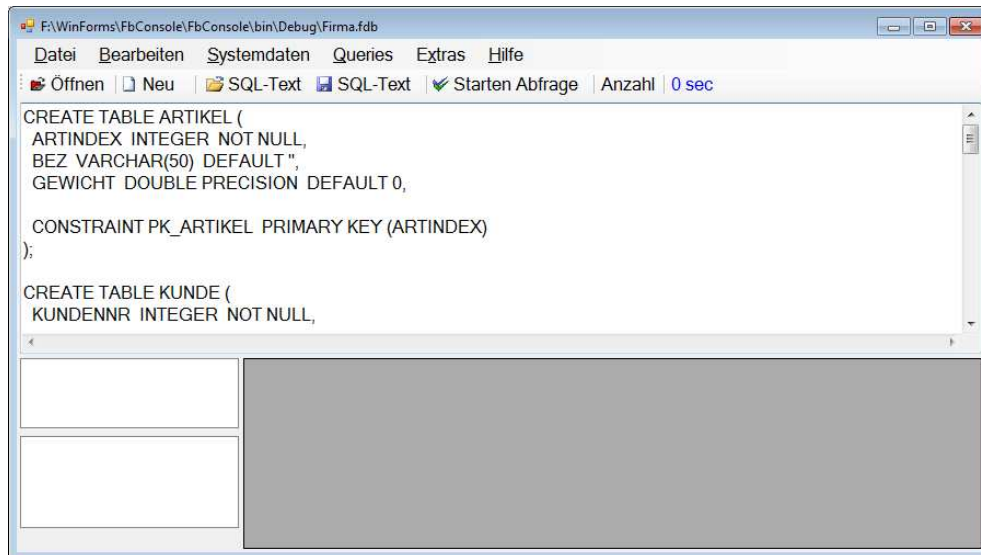


Abbildung 28 Eintragen der SQL-Skripte

Schalter „Starten Abfrage“

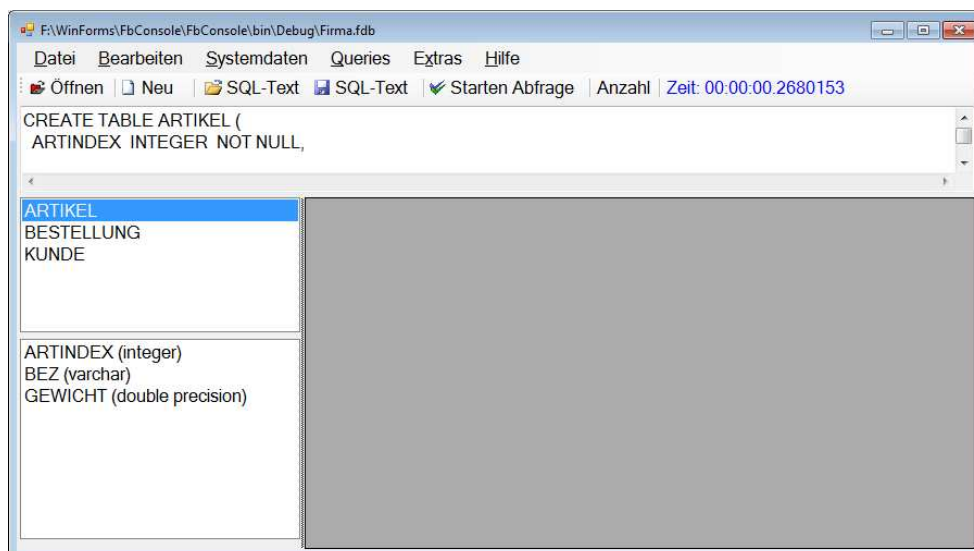


Abbildung 29 Anzeige der Datenbank-Tabellen

In der oberen Liste sind alle Entities eingetragen.

In der unteren Liste sind alle Spalten des aktuellen Entities eingetragen.

3.2 Daten eintragen

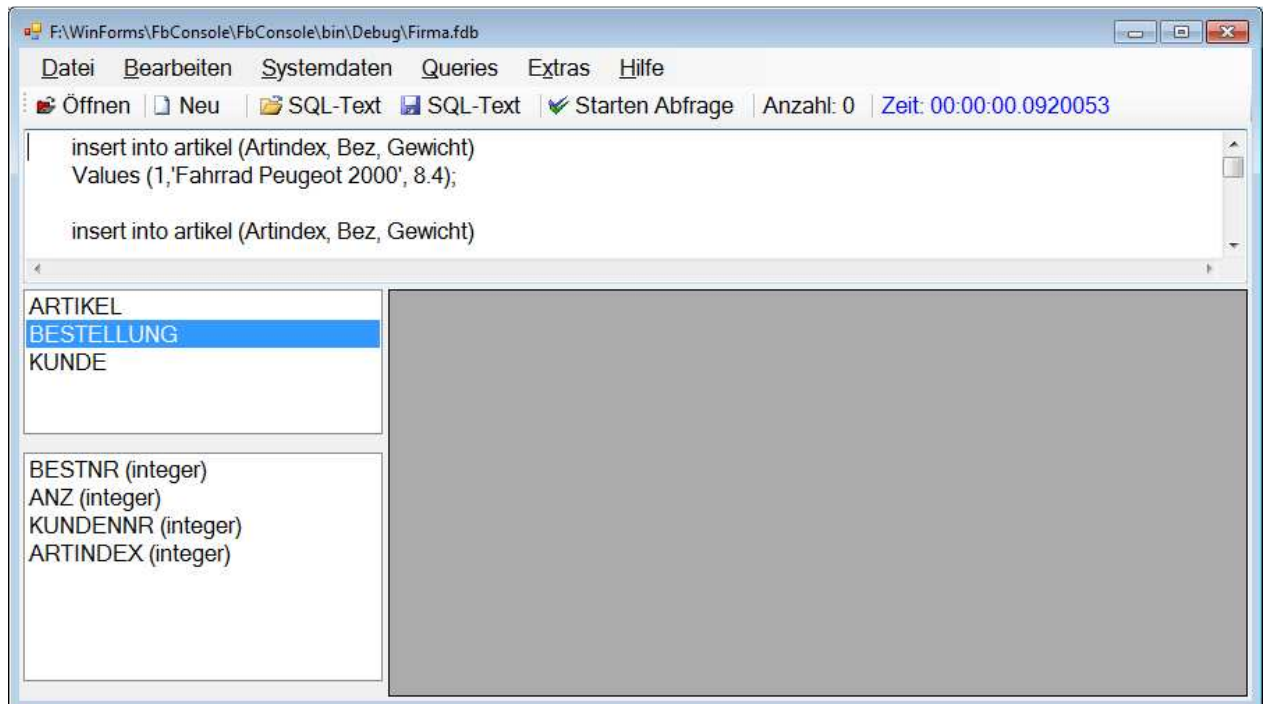


Abbildung 30 Daten mittels „Insert-Into“ eintragen

3.2.1 SQL-Befehle zum Eintragen:

```
insert into kunde (kundenr, name, vorname, strasse, plz, ort)
Values (101, 'Müller', 'Hans', 'Langer Weg 6', '01256',
'Weißenfels');
```

```
insert into kunde (kundenr, name, vorname, strasse, plz, ort)
Values (102, 'Schmidt', 'Andrea', 'Kastanienallee 42', '39114',
'Magdeburg');
```

```
insert into kunde (kundenr, name, vorname, strasse, plz, ort)
Values (103, 'Schmidt', 'Andrea', 'Kastanienallee 42', '39114',
'Magdeburg');
```

```
insert into kunde (kundenr, name, vorname, strasse, plz, ort)
Values (104, 'Meier', 'Antonia', 'Friedrichstraße 58', '38855',
'Wernigerode');
```

```
insert into artikel (Artindex, Bez, Gewicht)
Values (1, 'Fahrrad Peugeot 2000', 8.4);
```

```
insert into artikel (Artindex, Bez, Gewicht)
Values (2, 'Tablet Omega', 0.6);
```

```
insert into artikel (Artindex, Bez, Gewicht)
Values (3, 'Desktop HP 1200', 7.5);
```



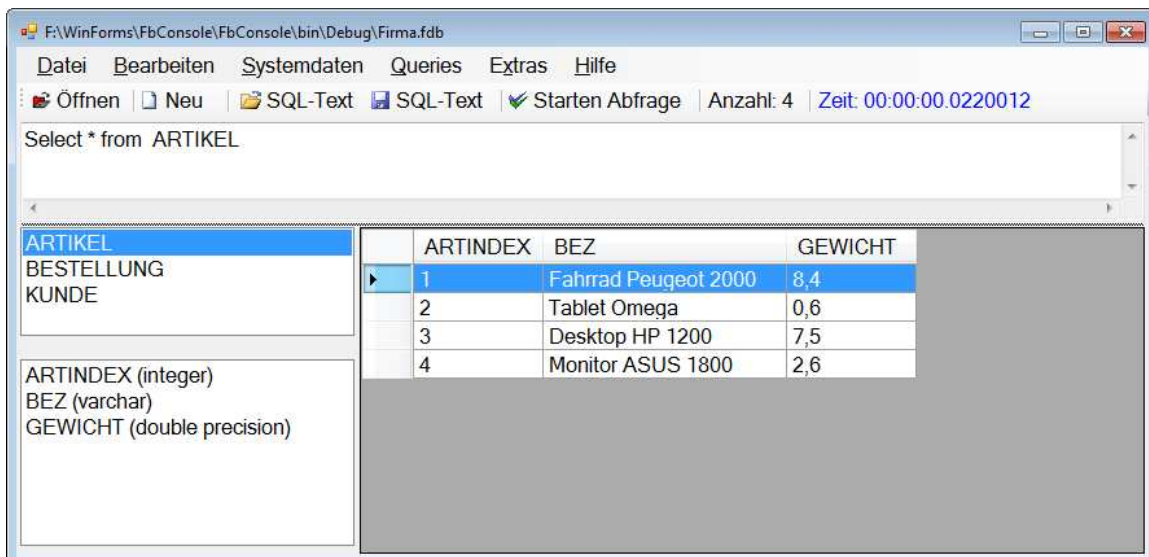
```
insert into artikel (Artindex, Bez, Gewicht)
Values (4,'Monitor ASUS 1800', 2.6);
```

```
insert into bestellung (bestnr, anz, kundenr, artindex)
Values (201, 2, 101, 1);
```

```
insert into bestellung (bestnr, anz, kundenr, artindex)
Values (202, 3, 104, 2);
```

```
insert into bestellung (bestnr, anz, kundenr, artindex)
Values (203, 1, 102, 4);
```

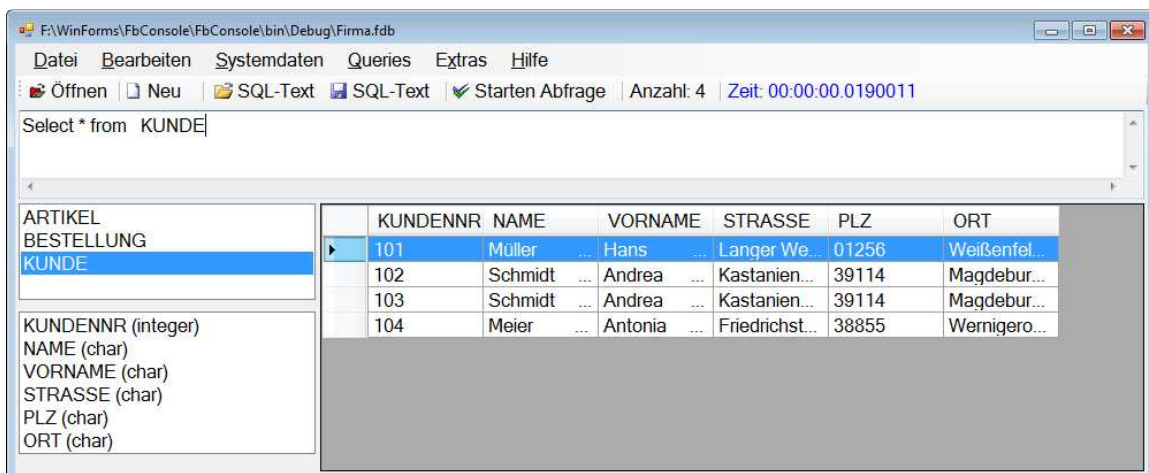
3.3 Testabfragen



The screenshot shows the FbConsole application window with the query "Select * from ARTIKEL" entered. The results are displayed in a table with columns ARTINDEX, BEZ, and GEWICHT. The table contains four rows of data. On the left, a list of tables (ARTIKEL, BESTELLUNG, KUNDE) is shown, and below it, the data types for the columns are listed: ARTINDEX (integer), BEZ (varchar), and GEWICHT (double precision).

ARTINDEX	BEZ	GEWICHT
1	Fahrrad Peugeot 2000	8,4
2	Tablet Omega	0,6
3	Desktop HP 1200	7,5
4	Monitor ASUS 1800	2,6

Abbildung 31 Abfrage bezüglich der Artikel



The screenshot shows the FbConsole application window with the query "Select * from KUNDE" entered. The results are displayed in a table with columns KUNDENNR, NAME, VORNAME, STRASSE, PLZ, and ORT. The table contains four rows of data. On the left, a list of tables (ARTIKEL, BESTELLUNG, KUNDE) is shown, and below it, the data types for the columns are listed: KUNDENNR (integer), NAME (char), VORNAME (char), STRASSE (char), PLZ (char), and ORT (char).

KUNDENNR	NAME	VORNAME	STRASSE	PLZ	ORT
101	Müller	Hans	Langer We...	01256	Weißentel...
102	Schmidt	Andrea	Kastanien...	39114	Magdebur...
103	Schmidt	Andrea	Kastanien...	39114	Magdebur...
104	Meier	Antonia	Friedrichst...	38855	Wernigero...

Abbildung 32 Abfrage bezüglich der Kunden

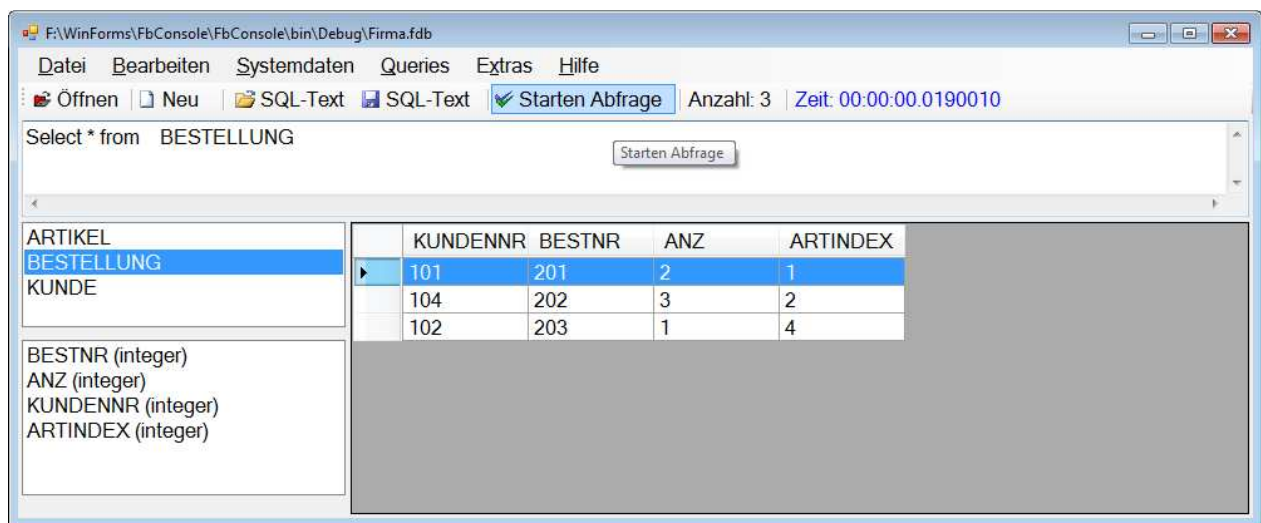


Abbildung 33 Abfrage bezüglich der Bestellungen

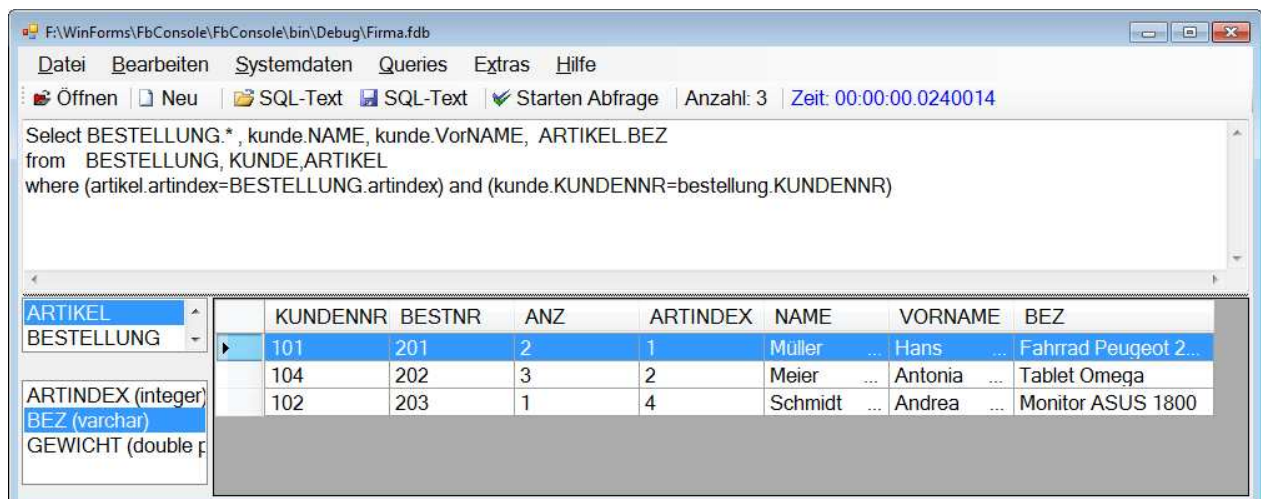


Abbildung 34 Abfrage bezüglich der Bestellungen

```
Select BESTELLUNG.* , kunde.NAME, kunde.VorNAME, ARTIKEL.BEZ
from BESTELLUNG, KUNDE,ARTIKEL
where (artikel.artindex=BESTELLUNG.artindex) and
      (kunde.KUNDENNR=bestellung.KUNDENNR)
```

4 Anhang

4.1 Generator

Generator definieren:

Um einen Generatoren zu erzeugen, ruft man das Menü „Einfügen“ mit dem Eintrag „Generator“ auf.

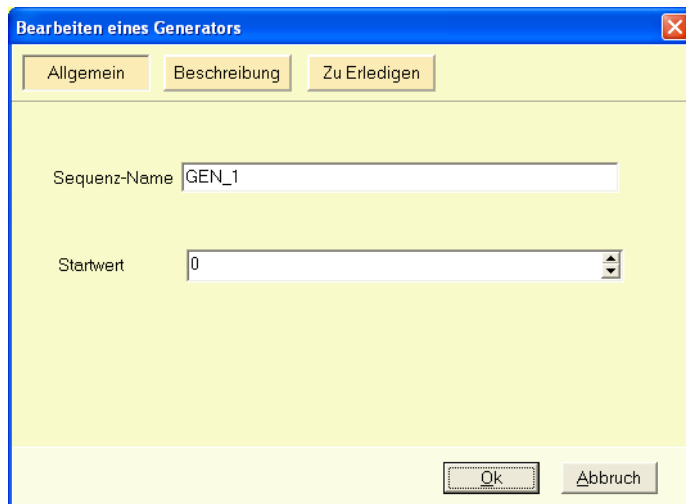


Abbildung 35 Definition eines Generators

In Firebird bzw. Interbase kann man neben dem Namen nur den Startwert definieren. Der erste richtige Wert ist dann um eins erhöht.

Manuelles Skript:

```
CREATE GENERATOR GEN_KUNDE ;  
SET GENERATOR GEN_KUNDE TO 0 ;
```

4.2 Sequenz

```
CREATE SEQUENCE <name>;  
ALTER SEQUENCE <name> RESTART WITH <value>;  
DROP SEQUENCE <name>;
```

```
SELECT NEXT VALUE FOR <SequenceName> FROM RDB$DATABASE;
```

```
<intvar> = NEXT VALUE FOR <SequenceName>;
```

4.3 Daten-Eintrag

Möchte man größere Datenmengen per Programmcode eintragen, so wird pro Datensatz eine Verbindung erstellen. Dies beeinträchtigt massiv die Performanz. Als Abhilfe kann man mehrere SQL-Anweisungen blocken.

Anfangswert: execute block as begin
Endwert: end

Beispiel:

```
execute block as begin
    insert into student (Matrnr, Nachname, Vorname, Findex)
    Values (12345, 'Müller', 'Andreas', 1);

    insert into student (Matrnr, Nachname, Vorname, Findex)
    Values (12346, 'Schmidt', 'Petra', 2);

    insert into student (Matrnr, Nachname, Vorname, Findex)
    Values (12347, 'Meier', 'Ralf', 3);

    insert into student (Matrnr, Nachname, Vorname, Findex)
    Values (12348, 'Brandt', 'Ute', 1);
end
```

Sinnvoll sind fünfzig bis hunderter Blöcke.

5 Indexverzeichnis

B

Beispiel eines logisches Modell	7
---------------------------------	---

D

DB-Designer	5
-------------	---

E

Eigenschaften des DB-Designer-Programms	5
Eigenschaften in Kurzform	5
Erstellen der SQL-Skripte	18

F

FBConsole	21
Daten eintragen	24
Insert into	24
Neue Datenbank	22
Testabfragen	25

G

Generator	27
Größe Datenmengen	28

I

Inhalt der ZIP-Datei	4
Insert into	28

L

Logisches Modell	6
------------------	---

N

Neue Datenbank	22
----------------	----

S

Sequenz	27
---------	----