

# FileMaker® 10

## ODBC- und JDBC-Handbuch



© 2004-2009 FileMaker, Inc. Alle Rechte vorbehalten.

FileMaker, Inc.  
5201 Patrick Henry Drive  
Santa Clara, California 95054, USA

FileMaker, das Dateiordner-Logo, Bento und das Bento-Logo sind in den USA und anderen Ländern entweder eingetragene Marken oder Marken von FileMaker, Inc. Mac und das Mac-Logo sind Eigentum von Apple Inc., in den USA und anderen Ländern eingetragen. Alle anderen Marken sind Eigentum der jeweiligen Inhaber.

Die FileMaker-Dokumentation ist urheberrechtlich geschützt. Sie dürfen diese Dokumentation ohne schriftliche Genehmigung von FileMaker weder vervielfältigen noch verteilen. Diese Dokumentation darf ausschließlich mit einer gültigen, lizenzierten Kopie der FileMaker-Software verwendet werden.

Alle in den Beispielen erwähnten Personen, Firmen, E-Mail-Adressen und URLs sind rein fiktiv und jegliche Ähnlichkeit mit bestehenden Personen, Firmen, E-Mail-Adressen und URLs ist rein zufällig. Die Danksagungen und Urheberrechtshinweise finden Sie im entsprechenden Dokument, das mit der Software geliefert wurde. Die Erwähnung von Produkten und URLs Dritter dient nur zur Information und stellt keine Empfehlung dar. FileMaker, Inc. übernimmt keine Verantwortung für die Leistung dieser Produkte.

Weitere Informationen finden Sie auf unserer Website unter [www.filemaker.de](http://www.filemaker.de).

# Inhalt

## Kapitel 1

### **Einführung**

Über dieses Handbuch	7
Verwenden einer FileMaker-Datenbank als Datenquelle	7
Zugriff auf eine bereitgestellte FileMaker Pro-Datenbank	8
Einschränkungen bei Verwendung von Tools Dritter	8
Netzwerkanforderungen	9
Aktualisieren von Dateien aus früheren Versionen	9
Installation aktueller Treiber	9

## Kapitel 2

### **Installation von FileMaker ODBC- und JDBC-Client-Treibern**

Lizenzbestimmungen	11
Hardware- und Software-Anforderungen	11
ODBC-Client-Treiber-Anforderungen (Windows)	11
ODBC-Client-Treiber-Anforderungen (Mac OS)	11
JDBC-Client-Treiber	12
Netzwerkanforderungen	12
ODBC-Client-Treiber-Installation (Windows)	12
ODBC-Client-Treiber-Installation (Mac OS)	13
JDBC-Client-Treiber-Installation (Windows und Mac OS)	13
Konfigurieren der Client-Treiber	14
Die nächsten Schritte	14

## Kapitel 3

### **Verwenden von ODBC zum gemeinsamen Verwenden von FileMaker-Daten**

Erläuterung von ODBC	15
Verwenden des ODBC-Client-Treibers	16
Übersicht über den Zugriff auf eine FileMaker-Datenbankdatei	17
Zugreifen auf eine FileMaker-Datenbankdatei von einer Windows-Anwendung aus	17
Festlegen der ODBC-Client-Treibereigenschaften für einen FileMaker-Datenquellennamen (Windows)	17
Überprüfen des Zugriffs über ODBC (Windows)	19
Zugreifen auf eine FileMaker-Datenbankdatei von einer Mac OS-Anwendung aus	20
Konfigurieren des ODBC-Client-Treibers (Mac OS)	20
Festlegen der ODBC-Client-Treibereigenschaften für einen FileMaker-Datenquellennamen (Mac OS)	20

## Kapitel 4

### **Verwenden von JDBC zum gemeinsamen Verwenden von FileMaker-Daten**

Erläuterung von JDBC	21
Verwenden des JDBC-Client-Treibers	21
Erläuterung des JDBC-Client-Treibers	21
Herstellen einer Verbindung zu Ihrer Datenbank mit einem JDBC-URL	22
Angaben von Treibereigenschaften im URL-Subnamen	24
Überprüfen des Zugriffs über JDBC	25

## Kapitel 5

### **Unterstützte Standards**

Unterstützung von Unicode-Zeichen	27
SQL-Anweisungen	27
SELECT-Anweisung	27
SQL-Klauseln	28
FROM-Klausel	29
WHERE-Klausel	29
GROUP BY-Klausel	29
HAVING-Klausel	30
UNION-Operator	30
ORDER BY-Klausel	30
FOR UPDATE-Klausel	31
DELETE-Anweisung	33
INSERT-Anweisung	33
UPDATE-Anweisung	34
CREATE TABLE-Anweisung	35
ALTER TABLE-Anweisung	35
CREATE INDEX-Anweisung	35
DROP INDEX-Anweisung	36
SQL-Statistikfunktionen:	36
SQL-Ausdrücke	37
Feldnamen	37
Konstanten und Literale	37
Exponentialschreibweise	38
Numerische Operatoren	38
Zeichenoperatoren	38
Datumsoperatoren	39
Relationale Operatoren	39
Logische Operatoren	40
Funktionen	41
Priorität der Operatoren	43
ODBC-Katalogfunktionen	43
JDBC-Metadaten-Funktionen	44
Reservierte SQL-Schlüsselwörter	45

Kapitel 6  
**Zuordnen von FileMaker-Feldern  
zu ODBC-Datentypen**

Kapitel 7  
**Zuordnen von FileMaker-Feldern  
zu JDBC-Datentypen**

Kapitel 8  
**ODBC- und JDBC-Fehlermeldungen**

ODBC-Fehlermeldungen	51
ODBC-Treiber-Fehlermeldungen	51
ODBC-Treiber-Manager-Fehlermeldungen	51
SequeLink Client-Fehlermeldungen	51
SequeLink Server-Fehlermeldungen	52
Datenquellen-Fehlermeldungen	52
JDBC-Fehlermeldungen	52
JDBC-Treiber-Fehlermeldungen	52
SequeLink Server-Fehlermeldungen	52
Datenquellen-Fehlermeldungen	53
<b>Index</b>	55



# Kapitel 1

## Einführung

Dieses Handbuch erläutert hilfreiche Konzepte und Details für den Austausch von FileMaker®-Daten mit anderen Anwendungen mithilfe von ODBC (Open Database Connectivity) und JDBC (Java Database Connectivity). Darüber hinaus dokumentiert das Handbuch, wie die ODBC- und JDBC-Client-Treiber bei Verwendung mit FileMaker Pro und FileMaker Server Advanced den Industriestandard für ODBC, JDBC und SQL (Structured Query Language) unterstützen.

Sie können FileMaker Pro, FileMaker Pro Advanced oder FileMaker Server Advanced verwenden, um Ihre Datenbanklösung zu erstellen und zu testen. Anschließend können Sie Ihre FileMaker-Datenbanklösung als Datenquelle mit ODBC- und JDBC-fähigen Anwendungen verwenden. Sie müssen die FileMaker ODBC- und JDBC-Treiber auf dem Rechner installieren, auf dem auch die Anwendung des Drittherstellers installiert ist.

**Wichtig** Dieses Handbuch beschreibt die Verwendung von FileMaker-Software als Datenquelle. Schrittweise Anleitungen zur Verwendung von FileMaker Pro als ODBC-Client-Anwendung erhalten Sie in der FileMaker Pro Hilfe.

Die folgende Tabelle bietet eine Übersicht über die Verwendungsweise von ODBC und JDBC mit FileMaker-Software.

Was wollen Sie erledigen?	Wie wollen Sie vorgehen?	Siehe
FileMaker Pro als Datenquelle verwenden/ FileMaker Pro-Daten gemeinsam nutzen	1. SQL-Queries 2. ODBC und JDBC	Dieses Handbuch
FileMaker Pro als ODBC-Client verwenden/auf ODBC-Daten zugreifen	1. Interaktiv über das Beziehungendiagramm 2. Einmalig statisch per ODBC-Import oder Datei (Windows) bzw. Ablage (Mac OS) > Öffnen. Außerdem der Scriptschritt „Datensätze importieren“ und der Scriptschritt „SQL ausführen“	FileMaker Pro Hilfe

## Über dieses Handbuch

- Informationen über die Verwendung von ODBC und JDBC mit früheren Versionen von FileMaker Pro erhalten Sie unter [www.filemaker.de/documentation](http://www.filemaker.de/documentation).
- Dieses Handbuch setzt voraus, dass Sie mit den Grundlagen der Verwendung von ODBC und JDBC und der Erstellung von SQL-Abfragen vertraut sind. Zusätzliche Informationen zu diesen Themen finden Sie in Büchern von Fremdanbietern.
- In diesem Handbuch bezieht sich „FileMaker Pro“ sowohl auf FileMaker Pro als auch auf FileMaker Pro Advanced. Ausgenommen ist die Beschreibung von Funktionen, die spezifisch für FileMaker Pro Advanced sind.

**Hinweis** Sie können PDF-Dateien der FileMaker-Dokumentation unter [www.filemaker.de/documentation](http://www.filemaker.de/documentation) herunterladen. Aktualisierungen dieses Dokuments erhalten Sie ebenfalls auf der Website.

## Verwenden einer FileMaker-Datenbank als Datenquelle

ODBC und JDBC sind Schnittstellen zur Programmierung von Anwendungen (APIs). Diese APIs geben Client-Anwendungen eine gemeinsame Sprache für die Interaktion mit einer Vielzahl von Datenquellen und Datenbankdiensten, einschließlich FileMaker Pro und FileMaker Server Advanced.

Alle Anwendungen, die ODBC und JDBC unterstützen, erkennen eine gemeinsame Untergruppe von SQL-Anweisungen. Wenn Sie mit SQL arbeiten, können Sie andere Anwendungen (wie Tabellenkalkulationen, Textverarbeitungsprogramme und Bericht-Tools) verwenden, um FileMaker-Daten anzuzeigen, zu analysieren und zu bearbeiten.

SQL wird über die ODBC- und JDBC-Schnittstellen an den FileMaker-Host der Datenquelle übergeben und führt Abfragen durch wie `SELECT vorname, nachname FROM kunden WHERE ort='Paris'` oder Aktualisierungen wie z. B. die Erstellung eines neuen Datensatzes mit `INSERT INTO kunden (vorname, nachname) VALUES ('Anna', 'Schmidt')`.

## Zugriff auf eine bereitgestellte FileMaker Pro-Datenbank

Sie können FileMaker Server Advanced und FileMaker Pro verwenden, um eine FileMaker-Datenbankdatei als Datenquelle bereitzustellen und Ihre Daten mit anderen Anwendungen auszutauschen, die ODBC und JDBC verwenden. Die folgende Tabelle beschreibt, welche Funktionen die einzelnen FileMaker-Produkte unterstützen.

Dieses FileMaker-Produkt	unterstützt
FileMaker Server Advanced	Bis zu 50 Verbindungen, lokale Zugriffe (auf demselben Computer) und Remote-Zugriffe (sowohl für Middleware wie Web-Server als auch für den Remote-Client-Zugriff über Desktop-Anwendungen)
FileMaker Pro	Bis zu neun Verbindungen und nur den lokalen Zugriff (auf demselben Computer)

Die ODBC- und JDBC-Plugin-Komponenten, die Sie für die gemeinsame Nutzung Ihrer Daten mit anderen Anwendungen benötigen, werden mit FileMaker Server Advanced und FileMaker Pro installiert.

Um auf eine bereitgestellte FileMaker-Datenbankdatei zuzugreifen, müssen Sie die entsprechenden ODBC- oder JDBC-Client-Treiber installieren. Installieren Sie den Client-Treiber auf dem Rechner, auf dem auch die Anwendung des Drittherstellers installiert ist. Weitere Informationen zur Installation von Treiberdateien für den Zugriff auf eine FileMaker-Datenquelle erhalten Sie in Kapitel 2, „Installation von FileMaker ODBC- und JDBC-Client-Treibern“.

Wenn Ihre FileMaker-Datenbanklösung mehr als eine FileMaker-Datenbankdatei verwendet, müssen sich alle Datenbankdateien auf dem gleichen Computer befinden.

Kapitel 5, „Unterstützte Standards“ beschreibt die SQL-Anweisungen, die von den ODBC- und JDBC-Client-Treibern unterstützt werden, wenn sie mit FileMaker Pro und FileMaker Server Advanced verwendet werden.

**Wichtig** Wenn Sie ODBC/JDBC-Sharing deaktivieren, nachdem es bereits aktiviert war, stehen von FileMaker Server Advanced oder FileMaker Pro zur Verfügung gestellte Daten sofort nicht mehr zur Verfügung. Der Datenbankadministrator kann ODBC- und JDBC-Client-Anwendungen nicht über die Verfügbarkeit der Datenquelle unterrichten (der Administrator kann nur mit Clients von FileMaker-Datenbankdateien kommunizieren). Es werden keine Fehler gemeldet und die Client-Anwendung sollte Benutzern mitteilen, dass die Datenquelle nicht verfügbar ist und Transaktionen nicht beendet werden können. Falls eine Client-Anwendung eine Verbindung zu einer nicht verfügbaren FileMaker-Datenbankdatei herstellen möchte, wird das Fehlschlagen der Verbindung durch eine Meldung angezeigt.

## Einschränkungen bei Verwendung von Tools Dritter

- Microsoft Abfrage-Assistent: In einer FileMaker-Datenquelle können Sie nicht auf Tabellen- oder Spaltentitel zugreifen, die High-ASCII- oder Doppelbyte-Zeichen enthalten. Verwenden Sie stattdessen Microsoft Query und geben Sie die Zeichen umschlossen von doppelten Anführungszeichen manuell ein.
- Microsoft Access: In einer FileMaker-Datenquelle können Sie nicht auf Tabellen- oder Spaltentitel zugreifen, die High-ASCII- oder Doppelbyte-Zeichen enthalten.

## Netzwerkanforderungen

Sie benötigen ein TCP/IP-Netzwerk, um mithilfe von FileMaker Server Advanced eine FileMaker-Datenbankdatei über ein Netzwerk als Datenquelle bereitzustellen. FileMaker Pro unterstützt nur den lokalen Zugriff (auf demselben Computer).

## Aktualisieren von Dateien aus früheren Versionen

### Installation aktueller Treiber

Wenn Treiber für Versionen vor FileMaker Server 9 Advanced oder FileMaker Pro 9 installiert sind, müssen Sie diese Treiber deinstallieren und den Treiber für Version 10 installieren. Weitere Informationen finden Sie in Kapitel 2, „Installation von FileMaker ODBC- und JDBC-Client-Treibern“.

**Hinweis** Sie müssen für jede FileMaker-Datenbankdatei, auf die Sie als Datenquelle zugreifen möchten, einen eigenen Datenquellennamen erstellen. Falls Sie den Zugriff vorher über einen einzelnen Datenquellennamen eingerichtet haben, der es ermöglicht, Tabellen auf mehrere FileMaker-Datenbankdateien zu verteilen, müssen Sie diese Tabellen jetzt in einer einzigen Datenbankdatei zusammenfassen (oder mehrere Datenquellennamen erstellen).



# **Kapitel 2**

## ***Installation von FileMaker ODBC- und JDBC-Client-Treibern***

Diese Anleitung unterstützt Sie bei der Installation der Treiber für den Zugriff auf eine FileMaker®-Datenquelle von anderen Anwendungen über ODBC (Open Database Connectivity) und JDBC (Java Database Connectivity). Die Client-Treiber stehen über eigene Installationsprogramme auf der FileMaker CD oder im elektronischen Download im Ordner \xDBC zur Verfügung. Die neuesten Versionen der Client-Treiber stehen ebenfalls unter [www.filemaker.de/support/technologies](http://www.filemaker.de/support/technologies) zur Verfügung.

Falls Sie eine FileMaker-Datenbankdatei mithilfe von FileMaker Server Advanced bereitstellen, müssen Sie den entfernten Anwendern die Client-Treiber zur Verfügung stellen.

Nach Installation des benötigten Client-Treibers können Sie den Treiber konfigurieren, um auf eine FileMaker-Datenquelle zuzugreifen und SQL (Structured Query Language)-Abfragen für die Arbeit mit Daten aufzubauen.

### **Lizenzbestimmungen**

Die ODBC- und JDBC-Client-Treiber sind die Treiberbestandteile der FileMaker-Software, die es Anwendungen Dritter oder von Ihnen entwickelten Anwendungen ermöglichen, auf FileMaker-Dateien als ODBC- oder JDBC-Datenquelle zuzugreifen.

### **Hardware- und Software-Anforderungen**

Um die ODBC- und JDBC-Client-Treiber zu installieren und zu verwenden, benötigen Sie die folgende Mindestausstattung:

#### **ODBC-Client-Treiber-Anforderungen (Windows)**

- Pentium 300 MHz oder schneller
- Insgesamt 64 MB RAM für Windows 2000, 128 MB RAM für Windows XP
- Microsoft Data Access Components (MDAC), 2.8 SP1
- Windows MDAC 6.0 für Vista

#### **ODBC-Client-Treiber-Anforderungen (Mac OS)**

- Apple G3 oder schneller (keine G3-Upgradekarten)
- 128 MB Gesamtarbeitsspeicher
- Mac OS X 10.3.9 oder 10.4 (Die Software kann auch mit späteren Versionen verwendet werden, die von FileMaker zertifiziert sind.)

## JDBC-Client-Treiber

Anwendungen mit	Benötigen
JDBC 1,22 API	JDK 1.2-kompatible Java Virtual Machine (JVM)
JDBC 2.0 Core API	JDK 1.3-kompatible JVM
JDBC 2.0 Optional Package	JDK 1.3-kompatible JVM Ferner werden folgende APIS benötigt und mit dem SequeLink-Java-Client geliefert: <ul style="list-style-type: none"> <li>▪ JDBC 2.0 Optional Package</li> <li>▪ JNDI 1.2</li> <li>▪ JTA 1.0.1</li> </ul>
JCA API	JDK 1,3
JDBC 3.0 API	JDK 1.4 oder 1.5

Um festzustellen, welche Java-Version Sie verwenden, öffnen Sie ein Befehlsfenster (Windows) bzw. Terminalfenster (Mac OS) und geben Sie `java-version` ein.

## Netzwerkanforderungen

Wenn Sie auf eine FileMaker-Datenquelle zugreifen werden, die auf einem anderen Computer bereitgestellt wird, benötigen Sie einen Netzwerkzugriff über TCP/IP.

## ODBC-Client-Treiber-Installation (Windows)

Wenn bereits ein ODBC-Client-Treiber für Windows installiert ist, deinstallieren Sie ihn über **Software**, bevor Sie eine aktuellere Version installieren.

### So installieren Sie den ODBC-Client-Treiber:

Sie benötigen MDAC 2.8 SP1 (verfügbar unter [www.microsoft.com](http://www.microsoft.com)), um die ODBC-Client-Treiber zu installieren.

1. Doppelklicken Sie im Ordner `\xDBC\ODBC Client Driver Installer` auf **setup**.

Der DataDirect SequeLink for ODBC 5,5 - InstallShield Wizard wird angezeigt.

2. Installieren Sie den ODBC-Client-Treiber, indem Sie den Bildschirmanweisungen folgen.

Das Data Source SyncTool, der Data Source SyncTool Administrator und SequeLink Online Books sind nicht als Teil der Standardinstallation ausgewählt. Sie benötigen sie nicht für die Verwendung des ODBC-Client-Treibers.

Standardmäßig wird der ODBC-Client-Treiber in diesem Ordner installiert:

`C:\Programme\DataDirect\slodbc55`. Wählen Sie bei Bedarf ein anderes Laufwerk oder einen anderen Ordner.

3. Wenn die Installation abgeschlossen ist, klicken Sie auf **Fertig stellen**.

Der ODBC-Client-Treiber, **DataDirect 32-BIT SequeLink 5,5**, steht jetzt für die Konfiguration für den Zugriff auf eine FileMaker-Datenquelle bereit.

**Hinweis** Wenn Sie bei der direkten Installation Probleme feststellen, installieren Sie den Treiber stattdessen über **Software** unter Windows (wählen Sie **Start > Systemsteuerung > Software**).

## ODBC-Client-Treiber-Installation (Mac OS)

Wenn Sie den ODBC-Client-Treiber für Mac OS bereits installiert haben, deinstallieren Sie ihn, bevor Sie eine aktualisierte Version installieren.

### So installieren Sie den ODBC-Client-Treiber:

Kopieren Sie die Datei `SequeLink.bundle` aus dem Ordner `/xDBC/ODBC Client Driver Installer` entweder in Ihre System- oder Benutzer-Library. Wenn es keinen Ordner `/ODBC` gibt, erstellen Sie einen solchen Ordner manuell:

Library	SequeLink.bundle in diesen Ordner kopieren:	Verwenden Sie diesen Treiber-Dateipfad bei der Konfiguration:
System	<code>/Library/ODBC</code>	<code>/Library/ODBC/SequeLink.bundle/Contents/MacOS/ivslk20.dylib</code>
Anwender	<code>/Benutzer/&lt;Benutzer&gt;/Library/ ODBC</code>	<code>/Benutzer/&lt;Benutzer&gt;/Library/ODBC/SequeLink.bundle/Contents/MacOS/ ivslk20.dylib</code>

**Hinweis** Wenn Sie mit einem Konto mit Administratorrechten angemeldet sind, sind keine Änderungen an den Berechtigungen erforderlich, nachdem Sie den `/ODBC`-Ordner erstellt haben. Der ODBC-Ordner muss über folgende Berechtigungen verfügen:

- Eigentümer root, admin-Gruppe
- Lese-/Schreib-/Ausführberechtigungen für root und admin
- Lese-/Ausführberechtigungen für andere Konten

Dieser Client-Treiber wurde mit ODBC Administrator 1.0.2 (verfügbar mit Mac OS 10.4.11) und ODBC Administrator 1.1 (verfügbar mit Mac OS 10.5.4) getestet.

**Wichtig** Verwenden Sie den ODBC Administrator, der mit der ODBC-Treiber-Manager-Software geliefert wurde und für Ihre Client-Anwendung empfohlen wird.

## JDBC-Client-Treiber-Installation (Windows und Mac OS)

Das Installationsprogramm und der JDBC-Client-Treiber funktionieren für Windows und Mac OS. Sie benötigen Schreibzugriff für den Ordner, in dem Sie den JDBC-Client-Treiber installieren. Standardmäßig ist der Ordner, der das Installationsprogramm enthält, der Installationsordner.

Wenn Sie den JDBC-Client-Treiber früher installiert haben, deinstallieren Sie ihn, bevor Sie eine aktualisierte Version installieren.

### So installieren Sie den JDBC-Client-Treiber:

1. Öffnen Sie den Ordner `\xDBC\JDBC Client Driver Installer` und doppelklicken Sie auf `sljcininstaller.jar`. Der DataDirect SequeLink for JDBC 5,5 - InstallShield Wizard wird angezeigt.

**Hinweis** Java Runtime Environment sollte .JAR-Archivdateien zugeordnet sein. Andere Anwendungen wie WinZip oder Stuffit können ebenfalls .JAR-Dateien zugeordnet sein. In diesem Fall können sie verhindern, dass JDBC Installer geöffnet wird. Wenn sich das Installationsfenster nicht öffnet, öffnen Sie ein Befehlsfenster (Windows) bzw. Terminalfenster (Mac OS) und wechseln Sie in den Ordner „JDBC Client Driver“. Aus diesem Ordner können Sie das Programm über folgenden Befehl starten: `java -jar sljcininstaller.jar`.

**2. Klicken Sie auf Weiter.**

Lesen Sie die Lizenzvereinbarung.

**3. Wenn Sie die Lizenzvereinbarung akzeptieren wollen, wählen Sie Lizenzvereinbarung akzeptieren und klicken Sie dann auf Weiter.**

Ein Fenster zeigt die Installationsoptionen an.

**4. Wählen Sie Entwicklerwerkzeuge installieren.**

Die Werkzeuge enthalten JDBCTest, das Sie bei der Überprüfung Ihrer JDBC-Verbindungen unterstützt.

**5. Geben Sie ein Installationsverzeichnis an.**

- Windows: Geben Sie einen Pfad ein, der Ihre Java-Executable-Datei (java.exe) enthält.
- Mac OS: Geben Sie /Library/Java/Extensions ein (oder einen anderen Speicherort, der im ClassPath Ihrer Java-Anwendung enthalten ist).

**6. Klicken Sie auf Weiter.**

**7. Bestätigen Sie Ihre Installationsauswahlen und klicken Sie auf Installieren.**

**8. Wenn die Installation abgeschlossen ist, klicken Sie auf Fertig stellen.**

Der JDBC-Client-Treiber steht jetzt für die Konfiguration für den Zugriff auf eine FileMaker-Datenquelle bereit.

## Konfigurieren der Client-Treiber

Bevor Sie eine Client-Anwendung verwenden können, um auf eine FileMaker-Datenquelle zuzugreifen, müssen Sie einen Client-Treiber für die Datenquelle konfigurieren. Die Konfigurationseinstellungen identifizieren den verwendeten Client-Treiber, den Speicherort der Datenquelle und Details zur Verbindung.

**Wichtig** Wenn Sie einen FileMaker-Client-Treiber konfigurieren, müssen Sie 2399 als Port angeben. Für ODBC (Windows) geben Sie den Port im ODBC Data Source Administrator an. Für ODBC (Mac OS) geben Sie den Port im ODBC Administrator an. Für JDBC geben Sie den Port im JDBC URL an.

Weitere Informationen über den ODBC-Client-Treiber unter Windows erhalten Sie über Windows Start > Programme > DataDirect Sequelink for ODBC 5,5 > Driver Help.

**Hinweis** Das Hilfesystem von DataDirect Technologies beschreibt einige Funktionen, die über den Umfang der Verwendung der ODBC- und JDBC-Client-Treiber für den Zugriff auf FileMaker-Datenquellen hinausgehen.

## Die nächsten Schritte

Nach Installation und Konfiguration eines Client-Treibers können Sie SQL-Abfragen aufbauen und ausführen, um auf eine FileMaker-Datenquelle zuzugreifen.

Client-Anwendungen verwenden für den Zugriff auf Datenquellen über ODBC gelegentlich eine andere Terminologie. Viele Anwendungen verfügen über Menüeinträge mit Namen wie Hole externe Daten oder SQL-Abfrage. Detaillierte Informationen hierzu finden Sie in der Dokumentation oder der Hilfe Ihrer Anwendung.

# Kapitel 3

## Verwenden von ODBC zum gemeinsamen Verwenden von FileMaker-Daten

Verwenden Sie den ODBC-Client-Treiber, um von einer anderen Anwendung aus eine Verbindung zu einer FileMaker-Datenquelle herzustellen. Die Anwendung, die den ODBC-Client-Treiber verwendet, kann direkt auf die Daten in einer FileMaker-Datenbankdatei zugreifen.

- Windows: Der FileMaker ODBC-Client-Treiber ist DataDirect 32-BIT SequeLink 5.5.
- Mac OS: Der FileMaker ODBC-Client-Treiber ist ivslk20.dylib.

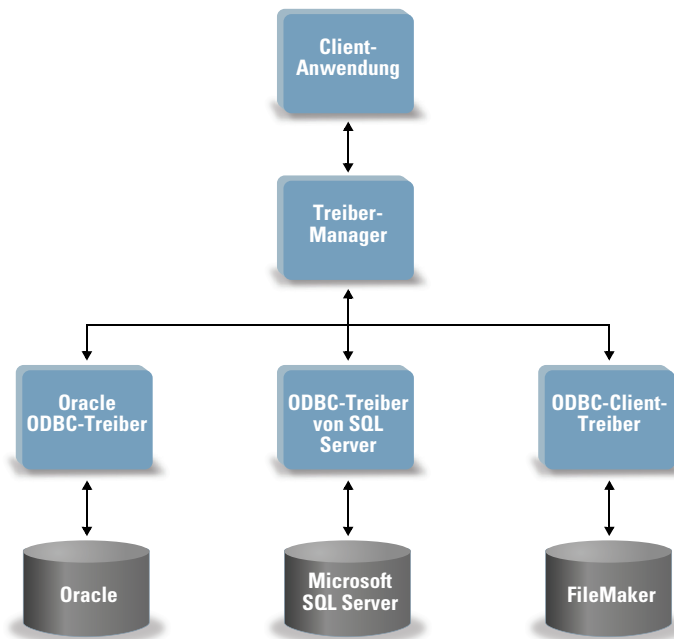
**Hinweis** Sie können auch FileMaker Pro als Client-Anwendung verwenden, die mithilfe von SQL über ODBC mit den Datensätzen einer anderen Datenquelle interagiert. Weitere Informationen über den Zugriff auf eine externe Datenquelle über ODBC finden Sie in der FileMaker Pro Hilfe.

### Erläuterung von ODBC

ODBC ist ein API, das Anwendungen den Zugriff auf Daten aus vielen Datenbankmanagementsystemen ermöglicht. ODBC gibt Client-Anwendungen eine gemeinsame Sprache für die Interaktion mit Datenquellen und Datenbankdiensten.

Alle Anwendungen, die ODBC unterstützen, erkennen eine gemeinsame Untergruppe von Anweisungen. Mithilfe von SQL können Sie andere Anwendungen (wie Tabellenkalkulationen, Textverarbeitungsprogramme und Bericht-Tools) verwenden, um FileMaker-Daten anzuzeigen, zu analysieren und zu bearbeiten (siehe Kapitel 5, „Unterstützte Standards“ hinsichtlich der vom ODBC-Client-Treiber unterstützten SQL-Anweisungen, -Funktionen und -Ausdrücke).

Ihre Anwendung kann über den ODBC-Client-Treiber direkt mit einer FileMaker-Datenbankdatei kommunizieren. Ihre SQL-Anweisungen werden an den FileMaker-Host der Datenbankdatei übergeben, und die Ergebnisse dieser Anweisungen werden an Sie zurückgesendet. Wenn Sie FileMaker Server Advanced verwenden, um eine FileMaker-Datenbankdatei als Datenquelle bereitzustellen, kann sich die Datenbankdatei auf einem anderen an Ihr Netzwerk angeschlossenen Rechner (dem Server) befinden, während sich die Client-Anwendung auf Ihrem Rechner (dem Client) befindet. Diese Verteilung wird als Client/Server-Konfiguration bezeichnet.



## Verwenden des ODBC-Client-Treibers

Sie können den ODBC-Client-Treiber mit jeder ODBC-kompatiblen Anwendung verwenden. Wenn Sie Ihre FileMaker-Datenbankdatei gemeinsam als Datenquelle verwenden, können Sie:

- Serienbriefe mit Microsoft Word erstellen,
- Diagramme mit Microsoft Excel erstellen,
- FileMaker-Daten zu einer DBMS wie Microsoft SQL Server verschieben,
- Ihre FileMaker-Daten mit Abfrage- oder Bericht-Tools weiter analysieren, um Diagramme zu zeichnen, ad-hoc-Abfragen zu erstellen und Tiefenanalysen durchzuführen und
- eine Microsoft Visual Basic-Anwendung entwickeln, die Informationen gemeinsam mit FileMaker Pro benutzt.

Wenn Sie eine FileMaker-Datenbankdatei als Datenquelle gemeinsam nutzen möchten, verwenden Sie FileMaker Pro, um Konten zu definieren, die auf die Datenbankdatei Zugriff haben sollen. Steuern Sie den Zugriff auf die Datenbankdatei, indem Sie den Konten Berechtigungen zuweisen, die das erweiterte Zugriffsrecht für den Zugriff über ODBC/JDBC mit einschließen. Als Letztes müssen Sie die FileMaker Server Advanced- oder FileMaker Pro-Host-Anwendung aktivieren, um Daten über ODBC/JDBC austauschen zu können. Detaillierte Informationen hierzu finden Sie in der FileMaker Pro- oder FileMaker Server Hilfe.

**Wichtig** Der ODBC-Client-Treiber ersetzt den FileMaker Pro ODBC-Treiber der vorherigen Version von FileMaker. Falls Sie den Zugriff auf eine FileMaker-Datenquelle zuvor mit dem alten Treiber eingerichtet haben, müssen Sie den Zugriff neu definieren, indem Sie den neuen Treiber verwenden und konfigurieren.

**Hinweis** Um zu bestätigen, dass der Sequelink Server aktuell läuft, können Sie unter Mac OS die Aktivitäts-Anzeige oder unter Windows den Task-Manager verwenden, um den Status des ddfmsoc-Prozesses zu prüfen. Wenn der Prozess gestartet wurde, heißt er FMSocket\_SL55 und Ereignisse werden mit diesem Namen protokolliert. Der Sequelink Server-Prozess läuft getrennt vom FileMaker Server-Prozess.

## Übersicht über den Zugriff auf eine FileMaker-Datenbankdatei

In einer ODBC-kompatiblen Anwendung können Sie SQL-Abfragen konstruieren, um auf eine FileMaker-Datenbankdatei zuzugreifen. Der ODBC-Client-Treiber muss auf dem Computer installiert sein, der die SQL-Abfrage generiert.

So greifen Sie auf eine FileMaker-Datenbankdatei zu:

1. Überprüfen Sie in FileMaker Pro die Berechtigungen, die Sie Konten für den Zugriff auf die Datenbankdatei zugewiesen haben.  
Konten, die auf die Datenbankdatei zugreifen sollen, müssen über das erweiterte Zugriffsrecht Zugriff über ODBC/JDBC verfügen.
2. Aktivieren Sie die FileMaker Server Advanced- (über FileMaker Server Admin Console) oder FileMaker Pro-Host-Anwendung, um Daten über ODBC/JDBC austauschen zu können.  
FileMaker Server Admin Console: Klicken Sie auf ODBC/JDBC und wählen Sie ODBC/JDBC aktivieren.  
FileMaker Pro: Wählen Sie Datei (Windows) bzw. Ablage (Max OS) > Sharing > ODBC/JDBC und stellen Sie ODBC/JDBC-Sharing auf Ein.
3. Vergewissern Sie sich, dass die FileMaker-Datenbankdatei, auf die Sie zugreifen möchten, bereitgestellt und verfügbar ist.  
Wenn Ihre FileMaker-Datenbanklösung mehr als eine FileMaker-Datenbankdatei verwendet, müssen sich alle Datenbankdateien auf dem gleichen Computer befinden.
4. Stellen Sie eine Verbindung zur FileMaker-Datenquelle her.
5. Konstruieren Sie in der Client-Anwendung eine SQL-Abfrage und führen Sie sie aus.  
Jede geöffnete und für den Zugriff eingerichtete FileMaker-Datenbankdatei ist eine separate Datenquelle (für jede FileMaker-Datenbankdatei, auf die Sie als Datenquelle zugreifen möchten, erstellen Sie einen DSN).  
Jede Datenbank kann über eine oder mehrere Tabellen verfügen. FileMaker-Felder werden als Spalten dargestellt. Der vollständige Feldname einschließlich etwaiger Sonderzeichen wird als Spaltenname angezeigt.

**Hinweis** In Windows kann Microsoft Access nur jeweils 32 oder weniger Felder über ODBC aus einer FileMaker Pro-Datenbankdatei importieren. Falls Ihre Datenbankdatei über mehr als 32 Felder verfügt, importieren Sie sie in 32er Schritten.

## Zugreifen auf eine FileMaker-Datenbankdatei von einer Windows-Anwendung aus

### Festlegen der ODBC-Client-Treibereigenschaften für einen FileMaker-Datenquellennamen (Windows)

Erstellen Sie für jede FileMaker-Datenbankdatei, auf die Sie als Datenquelle zugreifen möchten, einen eigenen Datenquellennamen. Der Datenquellename identifiziert den FileMaker ODBC-Client-Treiber, den Ort der FileMaker-Host-Anwendung und die FileMaker-Datenbankdatei, auf die Sie als Datenquelle zugreifen.

So richten Sie ODBC-Client-Treibereigenschaften ein oder ändern sie:

1. Öffnen Sie das Kontrollfeld „ODBC-Datenquellenadministrator“.

Wählen Sie in der Windows-Systemsteuerung Verwaltung > Datenquellen (ODBC).

Unter Windows XP wird Verwaltung in der Kategorie Leistung und Wartung angezeigt. Unter Windows Vista wird Verwaltung in der Kategorie System und Wartung angezeigt.

2. Klicken Sie auf das Register System-DSN.

Wenn Sie Ihre Datenquelle als Benutzer-DSN oder Datei-DSN einrichten, klicken Sie auf das entsprechende Register.

3. Klicken Sie auf Hinzufügen.

**Hinweis** Falls Sie die Eigenschaften einer bestehenden Datenquelle ändern, wählen Sie die Datenquelle, klicken auf Konfigurieren und fahren mit Schritt 6 fort.

4. Wählen Sie DataDirect 32-BIT SequeLink 5,5 aus der Treiberliste.

Wenn der Treiber nicht im ODBC Data Source Administrator aufgeführt wird, suchen Sie den Eintrag (Standard) im Registrierungsschlüssel

HKEY\_LOCAL\_MACHINE\SOFTWARE\ODBC\ODBCINST.INI\ODBC Drivers über Regedit. (Wählen Sie Start > Ausführen > Regedit). Die Daten für diesen Eintrag sollten (Wert nicht gesetzt) lauten. Wenn Sie stattdessen eine leere Zeichenfolge sehen, klicken Sie mit der rechten Maustaste auf (Standard), wählen Sie Löschen und klicken Sie als Bestätigung auf Ja. So wird dieser Eintrag auf (Wert nicht gesetzt) zurückgesetzt. Deinstallieren Sie Ihren ODBC-Client-Treiber und installieren Sie ihn dann neu, um ihn im ODBC Data Source Administrator anzuzeigen.

5. Klicken Sie auf Fertigstellen.

Das Dialogfeld „DataDirect SequeLink für ODBC-Setup“ wird geöffnet. Das Register Allgemein ist ausgewählt.

6. Geben Sie für Datenquellename einen Namen ein, der für andere Benutzer eine Bedeutung hat, die auf die FileMaker-Datenquelle zugreifen.

Eine zusätzliche Beschreibung ist optional.

Stellen Sie sicher, dass LDAP verwenden deaktiviert ist. Der ODBC-Client-Treiber beinhaltet keine Filter-.DLLs. Daher können Sie durch Klicken auf „Konvertieren“ keinen ODBC-Filter konfigurieren.

7. Geben Sie für SequeLink Server Host den Speicherort Ihrer Datenquelle ein.

Falls Sie eine Verbindung zu einer FileMaker-Datenbankdatei herstellen, die von FileMaker Pro auf Ihrem lokalen Rechner bereitgestellt wird, geben Sie localhost (oder 127.0.0.1) ein.

Falls Sie eine Verbindung zu einer FileMaker-Datenbankdatei herstellen, die von FileMaker Server Advanced über ein Netzwerk bereitgestellt wird, geben Sie die IP-Adresse von FileMaker Server ein.

8. Geben Sie für SequeLink Server Port den Wert 2399 ein.

**Wichtig** Wenn Sie eine FileMaker-Datenquelle konfigurieren, müssen Sie als SequeLink Server Port den Wert 2399 angeben.

9. Geben Sie für Server-Datenquelle den Dateinamen der FileMaker-Datenbankdatei ein, die Sie als Datenquelle verwenden (ohne Dateinamenerweiterung).

**10.** Falls Sie in der Host-Anwendung Sharing über ODBC/JDBC aktiviert haben, klicken Sie auf die Schaltfläche rechts neben **Server-Datenquelle**, um die Dateinamen der derzeit geöffneten FileMaker-Datenbankdateien anzuzeigen, aus denen Sie auswählen können.

Wenn Ihr Datenbankname Leerzeichen enthält, ersetzen Sie sie durch Escape-Zeichen %20. Beispiel: MEINE%20DATENBANK.

**11.** Klicken Sie auf **OK**, um Ihre Datenquelleninformationen zu sichern.

Wenn Sie eine andere FileMaker-Datenbankdatei gemeinsam verwenden, gehen Sie zurück zu Schritt 3 und richten die Datenbankdatei als Datenquelle ein.

**12.** Klicken Sie auf **OK**, um das Dialogfeld „ODBC-Datenquellenadministrator“ zu schließen.

### Überprüfen des Zugriffs über ODBC (Windows)

So überprüfen Sie, ob Sie den ODBC-Client-Treiber korrekt konfiguriert haben, um auf die FileMaker-Datenquelle zugreifen zu können:

**1.** Öffnen Sie das Kontrollfeld „ODBC-Datenquellenadministrator“.

Wählen Sie in der Windows-Systemsteuerung **Verwaltung > Datenquellen (ODBC)**.

Unter Windows XP wird **Verwaltung** in der Kategorie **Leistung und Wartung** angezeigt.

**2.** Klicken Sie auf das Register **System-DSN**.

Wenn Sie Ihre Datenquelle als **Benutzer-DSN** oder **Datei-DSN** einrichten, klicken Sie auf das entsprechende Register.

**3.** Wählen Sie die FileMaker-Datenquelle, die Sie zuvor konfiguriert haben.

Der ursprünglich eingegebene Datenquellenname wird unter **Name** angezeigt und **DataDirect 32-BIT SequeLink 5.5** wird als Treiber angezeigt.

**4.** Klicken Sie auf **Konfigurieren**.

Das Dialogfeld „DataDirect SequeLink für ODBC-Setup“ wird geöffnet.

**5.** Klicken Sie auf **Verbindung testen**.

Sie werden aufgefordert, Ihren FileMaker-Kontonamen (unter **Datenbankbenutzername**) und Ihr Passwort (unter **Datenbankpasswort**) einzugeben.

Falls die Verbindung OK ist, wird die Meldung **Verbindungstest erfolgreich** angezeigt. Falls die Verbindung fehlschlägt:

- Vergewissern Sie sich, dass die FileMaker-Datenbankdatei bereitgestellt und verfügbar ist.
- Aktualisieren oder korrigieren Sie die Verbindungsinformationen.
- Vergewissern Sie sich, dass Ihr FileMaker-Konto über das erweiterte Zugriffsrecht **Zugriff über ODBC/JDBC** verfügt.
- Überprüfen Sie, ob die FileMaker Pro- oder FileMaker Server-Host-Anwendung für Sharing über ODBC/JDBC eingerichtet wurde.

## Zugreifen auf eine FileMaker-Datenbankdatei von einer Mac OS-Anwendung aus

Wenn Sie eigene Anwendungen erstellen, müssen Sie Version 3.52.1 der ODBC-Headers und -Libraries verwenden. Mit Version 3.51 erstellte Anwendungen können den Client-Treiber eventuell nicht laden.

### Konfigurieren des ODBC-Client-Treibers (Mac OS)

Konfigurieren Sie den Client-Treiber mithilfe von ODBC Administrator, der zusammen mit der ODBC-Treibermanager-Software geliefert wird und für Ihre Client-Anwendung empfohlen wird.

Dieser Client-Treiber wurde mit ODBC Administrator 1.0.2 (verfügbar mit Mac OS 10.4.11) und ODBC Administrator 1.1 (verfügbar mit Mac OS 10.5.4) getestet.

Wenn Sie einen Client-Treiber konfigurieren, werden Sie aufgefordert, eine kurze Beschreibung und den Pfad zu der Treiberdatei anzugeben:

**Wenn Sie SequeLink.bundle in diese Library kopiert haben:**

**Verwenden Sie diesen Treiber-Dateipfad bei der Konfiguration:**

/Library/ODBC

/Library/ODBC/SequeLink.bundle/Contents/MacOS/ivslk20.dylib

/Benutzer/<Benutzer>/Library/ODBC

/Benutzer/<Benutzer>/Library/ODBC/SequeLink.bundle/Contents/MacOS/ivslk20.dylib

Mithilfe von ODBC Administrator können Sie auch Schlüsselwörter und einen Setup File-Pfad optional definieren, aber der Client-Treiber benötigt diese Informationen nicht.

### Festlegen der ODBC-Client-Treibereigenschaften für einen FileMaker-Datenquellennamen (Mac OS)

Erstellen Sie für jede FileMaker-Datenbankdatei, auf die Sie als Datenquelle zugreifen möchten, einen eigenen Datenquellennamen. Der Datenquellename identifiziert den FileMaker ODBC-Client-Treiber, den Ort der File-Maker-Host-Anwendung und die FileMaker-Datenbankdatei, auf die Sie als Datenquelle zugreifen.

**Wichtig** Der ODBC-Client-Treiber für Mac OS unterstützt ASCII-Zeichen des oberen Bereichs, Doublebyte- und japanische Zeichen in Datenbank- und Tabellennamen nicht. Wenn Ihre FileMaker-Datenbankdatei diese Zeichen verwendet, erstellen Sie eine zweite Datenbank und verwenden Sie nur ASCII-Zeichen für Datei- und Tabellennamen. Erstellen Sie in der zweiten Datenbank einen Datenquellen-Verweis, der auf die Daten in der ursprünglichen Datenbankdatei verweist. Stellen Sie beide Dateien mit ODBC/JDBC bereit, aber verwenden Sie die zweite Datenbankdatei, wenn Sie den DSN definieren.

Zusätzlich müssen Sie diese Schlüsselwortwerte für den DSN angeben:

Schlüsselwort	Wert
Host	Falls Sie eine Verbindung zu einer FileMaker-Datenbankdatei herstellen, die von FileMaker Pro auf Ihrem lokalen Rechner bereitgestellt wird, geben Sie <code>localhost</code> (oder <code>127.0.0.1</code> ) ein. Falls Sie eine Verbindung zu einer FileMaker-Datenbankdatei herstellen, die von FileMaker Server Advanced über ein Netzwerk bereitgestellt wird, geben Sie die IP-Adresse oder den Hostnamen von FileMaker Server ein.
Port	Geben Sie <code>2399</code> ein.
ServerDataSource	Geben Sie den Dateinamen der FileMaker-Datenbankdatei an, die Sie als Datenquelle verwenden (ohne Dateinamenerweiterung). Wenn Ihr Datenbankname Leerzeichen enthält, ersetzen Sie sie durch Escape-Zeichen <code>%20</code> . Beispiel: <code>ServerDataSource=MEINE%20DATENBANK</code> . Doppelbytezeichen werden nicht unterstützt.

# Kapitel 4

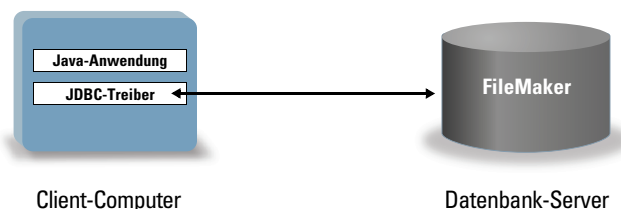
## Verwenden von JDBC zum gemeinsamen Verwenden von FileMaker-Daten

Wenn Sie ein Java-Programmierer sind, können Sie den JDBC-Client-Treiber mit jedem RAD- (Rapid Application Development) Tool verwenden, um Java-Anwendungen oder -Applets visuell zu erstellen, die eine Verbindung zu einer FileMaker-Datenquelle herstellen. Die Java-Anwendungen oder -Applets, die den JDBC-Client-Treiber verwenden, können direkt auf die Daten in einer FileMaker-Datenbankdatei zugreifen.

### Erläuterung von JDBC

JDBC ist eine Java-API zur Ausführung von SQL-Anweisungen, der Standardsprache für den Zugriff auf relationale Datenbanken. JDBC ist ein Name und kein Akronym, obwohl es praktisch für „Java Database Connectivity“ steht, das Java-Äquivalent für ODBC (Open Database Connectivity). JDBC ist als Low Level Interface (einfache Schnittstelle) ausgelegt, d. h. SQL-Befehle werden direkt über die Schnittstelle aufgerufen. Darüber hinaus dient es als Basis für Higher Level Interfaces (Schnittstellen mit mehr Bedienkomfort) und Tools.

Ihre Java-Applets oder -Anwendungen können über den JDBC-Client-Treiber direkt mit einer FileMaker-Datenbankdatei kommunizieren. Ihre SQL-Anweisungen werden an den FileMaker-Host der Datenbankdatei übergeben, und die Ergebnisse dieser Anweisungen werden an Sie zurückgesendet. Wenn Sie FileMaker Server verwenden, um eine FileMaker-Datenbankdatei als Datenquelle bereitzustellen, kann sich die Datenbankdatei auf einem anderen an Ihr Netzwerk angeschlossenen Rechner (dem Server) befinden, während sich die Java-Applets oder -Client-Anwendungen auf Ihrem Rechner (dem Client) befinden. Diese Verteilung wird als Client/Server-Konfiguration bezeichnet.



### Verwenden des JDBC-Client-Treibers

Der JDBC-Client-Treiber kann mit einem Java-Compiler bzw. RAD-Tool eine Verbindung zu Ihrer Datenbank herstellen, während Sie den Code für Ihre Java-Anwendung bzw. Ihr Java-Applet erstellen. Nachdem die Java-Anwendungen bzw. -Applets erstellt worden sind, muss der JDBC-Client-Treiber zusammen mit den Dateien zur Verfügung stehen oder in den Code integriert werden, damit die Anwendungen bzw. Applets mit der Datenbank kommunizieren können.

Um den JDBC-Client-Treiber nutzen zu können, muss die Java-Anwendung bzw. das Java-Applet den Treiber beim JDBC-Treiber-Manager registrieren, und Sie müssen in der Anwendung bzw. im Applet die korrekte JDBC-URL angeben. Die JDBC-URL wird benötigt, um die Verbindung zur Datenbank herstellen zu können.

### Erläuterung des JDBC-Client-Treibers

Der JDBC-Client-Treiber unterstützt teilweise die JDBC-3.0-Spezifikation. Folgende Funktionen werden von FileMaker nicht unterstützt:

- Sicherungspunkt-Unterstützung
- Abruf von automatisch erstellten Schlüsseln
- Übergabe von Parametern an ein aufrufbares Anweisungsobjekt nach Name
- Haltbarer Cursor, Unterstützung
- Interne Aktualisierung der Daten in BLOB- und CLOB-Objekten
- Abruf und Aktualisierung des von einem REF-Objekt referenzierten Objekts
- Aktualisierung von Spalten mit BLOB-, CLOB-, ARRAY- und REF-Datentypen
- Boolescher Datentyp
- DATALINK-Datentyp
- Umwandlung von Gruppen und Zuordnung von Typen
- Beziehung zwischen JDBC SPI und der Connector-Architektur

Weitere Informationen finden Sie unter [www.filemaker.de/support/technologies](http://www.filemaker.de/support/technologies). Zudem erhalten Sie in der *SequeLink Developer's Reference* unter [www.datadirect.com](http://www.datadirect.com) Informationen über die JDBC-Kompatibilität und Entwicklung von JDBC-Anwendungen für SequeLink.

Der JDBC-Client-Treiber wurde mit dem Java Development Kit (JDK) 1.5 (Mac OS) und 1.6 (Windows) getestet. Es handelt sich um einen Treiber vom Typ 4, einen Native Protocol All-Java-Treiber, der JDBC-Aufrufe direkt in das von FileMaker verwendete Netzwerkprotokoll konvertiert. Dieser Treibertyp bietet alle Vorteile von Java, u. a. automatische Installation (z. B. Herunterladen des JDBC-Treibers mit einem Applet, das den Treiber nutzt). Der Treiber unterstützt JDK 1.3 und Java 2, vorausgesetzt Sie verwenden nur JDBC 1.2-Aufrufe in einer Java 2-Umgebung.

Die Treiberklasse und Haupteinstiegswahl des Treibers:

`com.ddtek.jdbc.sequelink.SequeLinkDriver`

**Wichtig** Der JDBC-Client-Treiber ersetzt den FileMaker JDBC-Treiber der vorherigen Version von FileMaker. Falls Sie den Zugriff auf eine FileMaker-Datenquelle zuvor mit dem alten Treiber eingerichtet haben, müssen Sie den Zugriff neu definieren, indem Sie den neuen Treiber verwenden und konfigurieren.

**Hinweis** Um zu bestätigen, dass der SequeLink Server aktuell läuft, können Sie unter Mac OS die Aktivitäts-Anzeige oder unter Windows den Task-Manager verwenden, um den Status des `ddfmsoc`-Prozesses zu prüfen. Wenn der Prozess gestartet wurde, heißt er `FMSocket_SL55` und Ereignisse werden mit diesem Namen protokolliert. Der SequeLink Server-Prozess läuft getrennt vom FileMaker Server-Prozess.

## Herstellen einer Verbindung zu Ihrer Datenbank mit einem JDBC-URL

In Java erfolgt der Zugriff auf Ressourcen meist über URLs (Uniform Resource Locators). Eine JDBC-URL kennzeichnet die Datenbank, sodass der JDBC-Client-Treiber die Datenbank erkennen und eine Verbindung mit ihr herstellen kann.

Die JDBC-URL besteht aus drei Hauptkomponenten, die durch Doppelpunkt getrennt sind.

`jdbc:<Subprotokoll>:<Subname>`

Die erste Komponente des JDBC-URL ist immer das JDBC-Protokoll („`jdbc`“). Das *Subprotokoll* ist der Name des Treibers oder der Mechanismus, der mehrere Treiber unterstützt. Für den JDBC-Client-Treiber lautet das Subprotokoll `sequelink`. Der Subname ist die IP-Adresse des Computers, auf dem sich die FileMaker-Datenquelle befindet.

**Registrieren des JDBC-Client-Treibers und Herstellen einer Verbindung zu einer FileMaker-Datenquelle (Beispiel)**

Hier sehen Sie ein Snippet einer JDBC-Client-Anwendung, das:

1. den JDBC-Client-Treiber beim JDBC-Treiber-Manager registriert.
2. eine Verbindung zur FileMaker-Datenquelle (der JDBC-URL lautet jdbc:sequelink://17.184.17.170:2399) herstellt.
3. Fehlercodes ausgibt.

```
import java.sql.*;
class FMPJDBCTest
{
    public static void main(String[] args)
    {
        // Registrieren des JDBC-Client-Treibers
        try {
            Driver d =
            (Driver)Class.forName("com.ddtek.jdbc.sequelink.SequelinkDriver").newInstance();
        } catch(Exception e) {
            System.out.println(e);
        }
        // eine Verbindung mit FileMaker herstellen
        Connection con;
        try {
            con =
            DriverManager.getConnection("jdbc:sequelink://17.184.17.170:2399;
            user=ein Benutzer;password=ein Passwort;serverDataSource=Datenbank");
        } catch(Exception e) + ";serverdatasource=" + dbName{
            System.out.println(e);
        }
        // Verbindungswarnungen holen + ";serverDataSource=" + dbName
        SQLWarning warning = null;
        try {
            warning = con.getWarnings();
            if (warning == null) {
                System.out.println("Keine Warnungen");
                return;
            }
            while (warning != null) {
                System.out.println("Warnung: "+warning);
                warning = warning.getNextWarning();
            }
        } catch (Exception e) {
            Sysem.out.println(e);
        }
    }
}
}
```

**Hinweis** Dieses Beispiel ist nicht für die Kompilierung vorgesehen.

## Angeben von Treibereigenschaften im URL-Subnamen

Geben Sie im Subnamen des JDBC-URL die user- und password-Treibereigenschaften an. Diese Eigenschaften könnten auch an die Verbindung übergeben werden, wenn die Methode „DriverManager.getConnection“ über den Parameter „Properties“ aufgerufen wird.

- user: ein Konto in der FileMaker-Datenbankdatei, das Berechtigungen mit dem erweiterten Zugriffsrecht Zugriff über ODBC/JDBC verwendet
- password: das Passwort für das Konto in der FileMaker-Datenbankdatei

### Grundlegende JDBC-URL-Verbindung

Format: jdbc:sequelink://<SequeLink Host IP-Adresse>:<port>

Hierbei handelt es sich um die URL, mit der ohne Passwörter eine Verbindung zur FileMaker-Datenbankdatei hergestellt werden kann. Die Port-Nummer lautet immer 2399 (für JDBC-Sharing kann die Port-Nummer nicht geändert werden).

Wenn Sie die JDBC-URL-Verbindung auf demselben Rechner ausführen, auf dem sich die Datei befindet, können Sie jdbc:sequelink://localhost:2399 verwenden.

Beispiel: jdbc:sequelink://17.184.17.170:2399

### JDBC-URL-Verbindung mit in der URL definiertem Benutzernamen und Passwort

Format:

jdbc:sequelink://<SequeLink Host IP-Adresse>:<Port>;user=<Benutzername>;password=<Passwort>

Beispiel: jdbc:sequelink://17.184.17.170:2399;user=phil;password=jsp

### JDBC-URL-Verbindung mit in der URL angegebenem Datenbanknamen

Format:

jdbc:sequelink://<SequeLink Host IP-Adresse>:<Port>  
serverDataSource=<Datenbankname>

Beispiel: jdbc:sequelink://17.184.17.170:2399;serverDataSource=publications

Wenn Ihr Datenbankname Leerzeichen enthält, ersetzen Sie sie durch Escape-Zeichen %20.

Beispiel: jdbc:sequelink://17.184.17.170:2399;serverDataSource=MEINE%20DATENBANK

### JDBC-URL-Verbindung mit in der URL angegebenem Datenbanknamen, Benutzernamen und Passwort

Format 1 (Verwenden von Benutzername und Passwort des Datenspeichers): jdbc:sequelink://<Sequelink Host IP-Adresse>:<Port>;serverDataSource=<Datenbankname>;DBUser=<Datenbank-Benutzername>; DBPassword=<Datenbankpasswort>

Format 2 (Verwenden von Benutzername und Passwort des Host): jdbc:sequelink://<Sequelink Host IP-Adresse>:<Port>;serverDataSource=<Datenbankname>;HUser=<Benutzername>;HPassword=<Passwort>

Wenn Ihre FileMaker-Datenbanklösung mehrere FileMaker-Datenbankdateien verwendet, erstellen Sie eine zusätzliche Datenbankdatei, die alle notwendigen externen Datenquellenverweise, Tabellenauftritte und Beziehungen für Ihre Lösung enthält. Definieren Sie dann diese zusätzliche Datenbankdatei als Ihre Datenquelle in der JDBC URL. Sämtliche FileMaker-Datenbankdateien müssen sich auf dem gleichen Computer befinden.

**Hinweis** Secure Socket Layer (SSL)-Verschlüsselung wird nicht unterstützt. Um eine sicherere JDBC-Lösung zu erstellen, richten Sie eine Umgebung wie .ASP oder .JSP ein, in der der Web-Server die Daten über https bereitstellt und mit der FileMaker-Datenbankdatei kommuniziert, die durch eine Firewall abgesichert ist.

## Überprüfen des Zugriffs über JDBC

Wenn Sie den JDBC-Client-Treiber installieren, können Sie optional zusätzlich JDBCTest installieren, um Ihre Verbindungen zu überprüfen. Wählen Sie bei der Installation des JDBC-Client-Treibers Entwicklerwerkzeuge installieren, um JDBCTest zu installieren. Weitere Informationen hierzu finden Sie unter „JDBC-Client-Treiber-Installation (Windows und Mac OS)“ auf Seite 13.

Achten Sie auf folgende Punkte, wenn Sie den Zugriff auf eine FileMaker-Datenbankdatei über JDBC überprüfen:

- Die FileMaker-Datenbankdatei wird bereitgestellt und ist verfügbar.
- Ihr FileMaker-Konto verfügt über das erweiterte Zugriffsrecht Zugriff über ODBC/JDBC.
- Die FileMaker Pro- oder FileMaker Server Advanced-Host-Anwendung wurde für Sharing über ODBC/JDBC eingerichtet.

Wenn Sie eine FileMaker-Datenbankdatei als Datenquelle gemeinsam nutzen möchten, verwenden Sie FileMaker Pro, um Konten zu definieren, die auf die Datenbankdatei Zugriff haben sollen. Steuern Sie den Zugriff auf die Datenbankdatei, indem Sie den Konten Berechtigungen zuweisen, die das erweiterte Zugriffsrecht für den Zugriff über ODBC/JDBC mit einschließen. Als Letztes müssen Sie die FileMaker Server Advanced- oder FileMaker Pro-Host-Anwendung aktivieren, um Daten über ODBC/JDBC austauschen zu können. Weitere Informationen erhalten Sie in der FileMakerPro Hilfe.

- Die Registrierung des JDBC-Client-Treibers und die JDBC-URL sind korrekt (der Treiber kann sich innerhalb der Java-Anwendung oder auf dem Client-Computer befinden).

Weitere Informationen über die Verwendung von JDBC für die gemeinsame Nutzung von FileMaker-Daten erhalten Sie unter [www.filemaker.de/support/technologies](http://www.filemaker.de/support/technologies).



# Kapitel 5

## Unterstützte Standards

Dieses Kapitel beschreibt die SQL-Anweisungen und -Konstrukte, die von den FileMaker ODBC- und JDBC-Client-Treibern unterstützt werden. Verwenden Sie die Client-Treiber, um von einer Anwendung, die ODBC oder JDBC unterstützt, auf eine FileMaker-Datenbanklösung zuzugreifen. Die FileMaker-Datenbanklösung kann von FileMaker Pro oder FileMaker Server Advanced bereitgestellt sein.

Der ODBC-Client-Treiber unterstützt ODBC 3.5 Level 1 mit einigen Funktionen von Level 2. Der JDBC-Client-Treiber unterstützt teilweise die JDBC-3.0-Spezifikation. Weitere Informationen erhalten Sie unter [www.filemaker.de/support/technologies](http://www.filemaker.de/support/technologies) und in der *SequeLink Developer's Reference* unter [www.datadirect.com](http://www.datadirect.com). Die ODBC- und JDBC-Client-Treiber richten sich nach SQL-92-Entry-Level mit einigen zwischengeschalteten SQL-92-Funktionen.

### Unterstützung von Unicode-Zeichen

Die ODBC- und JDBC-Client-Treiber unterstützen die Unicode-API. Wenn Sie jedoch eine eigene Anwendung erstellen, die die Client-Treiber verwendet, benutzen Sie für Feldnamen, Tabellennamen und Dateinamen ASCII-Code (falls Nicht-Unicode-Abfrage-Tools oder -Anwendungen verwendet werden).

**Hinweis** Um Unicode-Daten einzufügen und abzurufen, verwenden Sie SQL\_C\_WCHAR. (Der SQL\_C\_BINARY-Datentyp wird nicht unterstützt.)

### SQL-Anweisungen

Die ODBC- und JDBC-Client-Treiber unterstützen folgende SQL-Anweisungen:

SELECT (siehe unten)	DELETE (Seite 33)	INSERT (Seite 33)	UPDATE (Seite 34)
CREATE TABLE (Seite 35)	ALTER TABLE (Seite 35)	CREATE INDEX (Seite 35)	DROP INDEX (Seite 36)

Die Client-Treiber unterstützen darüber hinaus die Zuordnung von FileMaker-Datentypen zu ODBC SQL- und JDBC SQL-Datentypen. Siehe Anhang 6, „Zuordnen von FileMaker-Feldern zu ODBC-Datentypen“, und Anhang 7, „Zuordnen von FileMaker-Feldern zu JDBC-Datentypen“, für Datentypkonvertierungen. Zusätzliche Informationen über das Erstellen von SQL-Abfragen finden Sie in Büchern von Fremdanbietern.

**Hinweis** Die ODBC- und JDBC-Client-Treiber erkennen nur die erste Wiederholung in einem Wiederholfeld. Außerdem unterstützen die Treiber keine Ausschnitte in FileMaker Pro.

### SELECT-Anweisung

Verwenden Sie die SELECT-Anweisung, um anzugeben, welche Spalten Sie anfordern. Geben Sie nach der SELECT-Anweisung den Spaltenausdruck (ähnlich wie Feldnamen) an, den Sie abrufen möchten (z. B. `last_name`). Ausdrücke können Rechenoperationen oder Zeichenfolgen beinhalten (z. B. `SALARY * 1,05`).

Die SELECT-Anweisung kann verschiedene Klauseln verwenden:

```
SELECT [DISTINCT] { * | spaltenausdruck [[AS] spaltenalias], ... }
FROM tabellenname [tabellenalias], ...
[ WHERE ausdr1 rel_operator ausdr2 ]
[ GROUP BY { spaltenausdruck, ... } ]
[ WHERE ausdr1 rel_operator ausdr2 ]
[ UNION [ALL] (SELECT...) ]
[ ORDER BY { sortierausdruck [DESC | ASC] }, ... ]
[ FOR UPDATE [OF { spaltenausdruck, ... } ] ]
```

Objekte in Klammern sind optional.

**Hinweis** SELECT \* für größere Datenbanken und SELECT-Anweisungen, die Tabellenaliasse oder Literale in der Projektionsliste verwenden, funktionieren eventuell nicht richtig. Um mögliche Verwirrungen zu vermeiden, vermeiden Sie Wildcards und geben Sie Tabellen- und Spaltennamen ohne Aliasse an.

spaltenalias kann verwendet werden, um der Spalte einen beschreibenden Namen zu geben oder um einen längeren Spaltennamen abzukürzen. Weisen Sie z. B. den Alias abteilung der Spalte abt zu:

```
SELECT abt AS abteilung FROM ang
```

Feldnamen kann der Tabellenname oder der Tabellenalias vorangestellt werden. Zum Beispiel ANG.NACHNAME oder A.NACHNAME, wobei A der Alias für die Tabelle ANG ist.

Der Operator DISTINCT kann dem ersten Spaltenausdruck vorangestellt werden. Dieser Operator eliminiert doppelte Reihen aus dem Ergebnis einer Abfrage. Beispiel:

```
SELECT DISTINCT abt FROM ang
```

## SQL-Klauseln

Die ODBC- und JDBC-Client-Treiber unterstützen folgende SQL-Klauseln:

### Verwenden Sie diese SQL-Klausel

### Um

FROM (siehe unten)	anzuzeigen, welche Tabellen in der SELECT-Anweisung verwendet werden
WHERE (Seite 29)	die Bedingungen anzugeben, die Datensätze erfüllen müssen, um abgefragt zu werden (wie FileMaker Pro-Suchabfragen)
GROUP BY (Seite 29)	die Namen eines oder mehrerer Felder anzugeben, nach denen die Ergebniswerte gruppiert werden sollen. Diese Klausel wird verwendet, um ein Set von Sammelwerten zurückzugeben, indem eine Zeile für jede Gruppe zurückgegeben wird (wie ein FileMaker Pro-Zwischenergebnis)
HAVING (Seite 30)	die Bedingungen für Gruppen von Datensätzen anzugeben (z. B. um nur die Abteilungen anzuzeigen, die Gehälter von insgesamt mehr als 200.000 Euro haben). Diese Klausel ist nur gültig, wenn Sie bereits eine GROUP BY-Klausel definiert haben.
UNION (Seite 29)	die Ergebnisse zweier oder mehrerer SELECT-Anweisungen in einem einzigen Ergebnis zu kombinieren.
ORDER BY (Seite 30)	anzuzeigen, wie die Datensätze sortiert sind.
FOR UPDATE (Seite 31)	„Positioned Updated“ und „Positioned Deletes“ über SQL-Cursor durchzuführen

**Hinweis** Wenn Sie versuchen, Daten von einer Tabelle ohne Spalten abzurufen, schlägt die SELECT-Anweisung fehl.

## FROM-Klausel

Die FROM-Klausel zeigt die Tabellen an, die in der SELECT-Anweisung verwendet werden. Das Format ist:

```
FROM tabellennamen [tabellenalias]
```

tabellennamen kann ein oder mehrere einfache Tabellennamen im aktuellen Arbeitsverzeichnis oder vollständige Pfadnamen sein.

tabellenalias kann verwendet werden, um der Tabelle einen beschreibenden Namen zu geben oder um einen längeren Tabellennamen abzukürzen.

Der Tabellennamen oder der Tabellenalias kann Feldnamen vorangestellt werden. Zum Beispiel können Sie mit der Tabellenspezifikation FROM angestellte A das Feld NACHNAME als A.NACHNAME angeben. Tabellenaliasse müssen verwendet werden, wenn die SELECT-Anweisung eine Tabelle mit sich selbst verknüpft. Beispiel:

```
SELECT * FROM angestellte A, angestellte F WHERE A.managernr =
F.angestelltennr
```

Das Gleichheitszeichen (=) nimmt nur passende Zeilen in die Ergebnisse auf.

Wenn Sie mehr als eine Tabelle verknüpfen und alle Zeilen auslassen möchten, die nicht in beiden Quelltabellen über entsprechende Zeilen verfügen, können Sie INNER JOIN verwenden. Beispiel:

```
SELECT *
FROM Verkaeufer INNER JOIN Verkaufsdaten
ON Verkaeufer.Verkaeuferrnr = Verkaufsdaten.Verkaeuferrnr
```

**Hinweis** OUTER JOIN wird zurzeit nicht unterstützt.

## WHERE-Klausel

Die WHERE-Klausel gibt die Bedingungen an, die Datensätze erfüllen müssen, um abgerufen zu werden. Die WHERE-Klausel enthält Bedingungen in der Form:

```
WHERE ausdr1 rel_operator ausdr2
```

ausdr1 und ausdr2 können Feldnamen, Konstantenwerte oder Ausdrücke sein.

rel\_operator ist der relationale Operator, der die beiden Ausdrücke verbindet. Die folgende SELECT-Anweisung ruft zum Beispiel die Namen der Angestellten ab, die ein Gehalt von mehr als 20.000 Euro haben.

```
SELECT nachname,vorname FROM ang WHERE gehalt >= 20000
```

**Hinweis** Wenn Sie vollständig qualifizierte Namen in der SELECT-Liste (Projektion) verwenden, müssen Sie auch vollständig qualifizierte Namen in der zugehörigen WHERE-Klausel verwenden.

## GROUP BY-Klausel

Die GROUP BY-Klausel gibt die Namen eines oder mehrerer Felder an, nach denen die Ergebniswerte gruppiert werden sollen. Diese Klausel wird verwendet, um eine Menge von Aggregatwerten zurückzugeben. Sie hat folgendes Format:

```
GROUP BY spaltenausdruecke
```

spaltenausdruecke muss mit dem in der SELECT-Anweisung verwendeten Spaltenausdruck übereinstimmen. Ein Spaltenausdruck kann ein oder mehrere Feldnamen der Datenbanktabelle, getrennt durch Kommata, oder ein oder mehrere Ausdrücke, getrennt durch Kommata, sein.

Das folgende Beispiel summiert die Gehälter jeder Abteilung.

```
SELECT abtnr, SUM (gehalt) FROM ang GROUP BY abtnr
```

Diese Anweisung gibt für jede Abteilungsnummer eine Zeile zurück. Jede Zeile enthält die Abteilungsnummer und die Summe der Gehälter der Mitarbeiter in der Abteilung.

## HAVING-Klausel

Die HAVING-Klausel ermöglicht Ihnen, die Bedingungen für Gruppen von Datensätzen anzugeben (z. B. um nur die Abteilungen anzuzeigen, die Gehälter von insgesamt mehr als 200.000 Euro haben). Diese Klausel ist nur gültig, wenn Sie bereits eine GROUP BY-Klausel definiert haben. Sie hat folgendes Format:

```
HAVING ausdr1 rel_operator ausdr2
```

ausdr1 und ausdr2 können Feldnamen, Konstantenwerte oder Ausdrücke sein. Diese Ausdrücke müssen nicht mit einem Spaltenausdruck in der SELECT-Klausel übereinstimmen.

rel\_operator ist der relationale Operator, der die beiden Ausdrücke verbindet. Das folgende Beispiel gibt nur die Abteilungen zurück, deren Gehaltssummen größer als 200.000 Euro sind.

```
SELECT abtnr, SUM (gehalt) FROM ang
GROUP BY abtnr HAVING SUM (gehalt) > 200000
```

## UNION-Operator

Der UNION-Operator kombiniert die Ergebnisse von zwei oder mehr SELECT-Anweisungen in ein einziges Ergebnis. Das einzelne Ergebnis besteht aus den zurückgegebenen Datensätzen der SELECT-Anweisungen. Standardmäßig werden doppelte Datensätze nicht zurückgegeben. Um doppelte Datensätze zurückzugeben, verwenden Sie das Schlüsselwort ALL (UNION ALL). Das Format ist:

```
SELECT anweisung UNION [ALL] SELECT anweisung
```

Bei Verwendung des UNION-Operators müssen die Auswahllisten für jede SELECT-Anweisung die gleiche Anzahl an Spaltenausdrücken mit den gleichen Datentypen besitzen und in der gleichen Reihenfolge angegeben sein. Beispiel:

```
SELECT nachname, gehalt, einstdatum FROM ang UNION SELECT name, zahlung,
geburtsdatum FROM person
```

Dieses Beispiel hat die gleiche Anzahl an Spaltenausdrücken und jeder Spaltenausdruck in der Reihenfolge hat den gleichen Datentyp.

Das folgende Beispiel ist nicht gültig, da sich die Datentypen der Spaltenausdrücke unterscheiden (GEHALT von ANG hat einen anderen Datentyp als NACHNAME von ERHÖHUNGEN). Dieses Beispiel hat die gleiche Anzahl an Spaltenausdrücken in jeder SELECT-Anweisung, aber die Ausdrücke erscheinen nach Datentyp nicht in der gleichen Reihenfolge.

```
SELECT nachname, gehalt FROM ang UNION SELECT gehalt, nachname FROM
erhoehungen
```

## ORDER BY-Klausel

Die ORDER BY-Klausel zeigt an, wie die Datensätze zu sortieren sind. Das Format ist:

```
ORDER BY {sort_ausdruck[DESC | ASC]}, ...
```

sort\_ausdruck können Feldnamen, Ausdrücke oder die Positionsnummer des zu verwendenden Spaltenausdrucks sein. Standard ist die Durchführung einer aufsteigenden (ASC) Sortierung.

Um zum Beispiel zuerst nach nachname und dann nach vorname zu sortieren, können Sie eine der folgenden SELECT-Anweisungen verwenden:

```
SELECT angnr, nachname, vorname FROM ang ORDER BY nachname, vorname
oder
```

```
SELECT angnr, nachname, vorname FROM ang ORDER BY 2,3
```

Im zweiten Beispiel ist nachname der zweite Spaltenausdruck nach SELECT, sodass ORDER BY 2 nach nachname sortiert.

## FOR UPDATE-Klausel

Die FOR UPDATE-Klausel führt „Positioned Updates“ oder „Positioned Deletes“ über SQL-Cursor durch. Das Format ist:

```
FOR UPDATE [OF spaltenausdruecke]
```

spaltenausdruecke ist eine Liste von Feldnamen in der Datenbanktabelle, die Sie aktualisieren möchten, getrennt durch ein Komma. spaltenausdruecke ist optional.

Das folgende Beispiel gibt alle Datensätze in der Angestellten-Datenbank zurück, die einen GEHALT-Feldwert von mehr als 20.000 Euro besitzen. Wenn jeder Datensatz abgerufen wird, wird er gesperrt. Wird der Datensatz aktualisiert oder gelöscht, wird die Sperre gehalten, bis Sie die Änderung bestätigen. Ansonsten wird die Sperre freigeben, wenn Sie den nächsten Datensatz abrufen.

```
SELECT * FROM ang WHERE gehalt > 20000 FOR UPDATE OF nachname, vorname,
    gehalt
```

### Weitere Beispiele:

Verwendung	Beispiel-SQL
Textkonstante	SELECT 'KatzeHund' FROM Verkaeufuer
Zahlenkonstante	SELECT 999 FROM Verkaeufuer
Datumskonstante	SELECT DATE '2004-06-05' FROM Verkaeufuer
Zeitkonstante	SELECT TIME '02:49:03' FROM Verkaeufuer
Zeitstempelkonstante	SELECT TIMESTAMP '2004-06-05 02:49:03' FROM Verkaeufuer
Textspalte	SELECT Firmenname FROM Verkaufsdaten SELECT DISTINCT Firmenname FROM Verkaufsdaten
Zahlenspalte	SELECT Betrag FROM Verkaufsdaten SELECT DISTINCT Betrag FROM Verkaufsdaten
Datumsspalte	SELECT Verkaufsdatum FROM Verkaufsdaten SELECT DISTINCT Verkaufsdatum FROM Verkaufsdaten
Zeitspalte	SELECT Verkaufszeit FROM Verkaufsdaten SELECT DISTINCT Verkaufszeit FROM Verkaufsdaten
Zeitstempelspalte	SELECT Verkaufszeitstempel FROM Verkaufsdaten SELECT DISTINCT Verkaufszeitstempel FROM Verkaufsdaten
BLOB <sup>a</sup> -Spalte	SELECT Firmenbroschueren FROM Vertriebsdaten SELECT GETAS(Firmenlogo, 'JPEG') FROM Vertriebsdaten
Jokerzeichen *	SELECT * FROM Verkaeufuer SELECT DISTINCT * FROM Verkaeufuer

a. Ein BLOB ist ein FileMaker-Datenbankdatei-Medienfeld.

### Hinweise zu den Beispielen

Eine Spalte ist ein Verweis auf ein Feld in der FileMaker-Datenbankdatei (das Feld kann viele unterschiedliche Werte enthalten).

Das Jokerzeichen (\*) ist eine Abkürzung für „Alles“. Für das Beispiel `SELECT * FROM Verkaeufuer` ist das Ergebnis alle Zeilen in der Tabelle `Verkaeufuer`. Für das Beispiel `SELECT DISTINCT * FROM Verkaeufuer` ist das Ergebnis alle eindeutigen Zeilen in der Tabelle `Verkaeufuer` (keine doppelten Werte).

**Hinweis** `SELECT *` Anweisungen für größere Datenbanken funktionieren eventuell nicht richtig. Um mögliche Verwirrungen zu vermeiden, vermeiden Sie Wildcards und geben Sie Tabellen- und Spaltennamen ohne Aliasse an.

### Abrufen des Inhalts eines Medienfelds: CAST()-Funktion und GetAs()-Funktion

Sie können Binärdaten, Dateiverweisinformationen oder Daten eines angegebenen Dateityps von einem Medienfeld abrufen.

Um Binärdaten abzurufen, verwenden Sie eine Standard-SELECT-Anweisung. Beispiel:

```
SELECT Firmenbroschueren FROM Vertriebsdaten
```

Wenn Datei- oder JPEG-Daten existieren, ruft die SELECT-Anweisung die Daten in Binärform ab. Andernfalls gibt die SELECT-Anweisung <null> zurück.

Um Dateiverweisinformationen von einem Medienfeld wie den Dateipfad zu einer Datei, einem Bild oder Quicktime-Film abzurufen, verwenden Sie die CAST-Funktion mit einer SELECT-Anweisung. Beispiel:

```
SELECT CAST(Firmenbroschueren AS VARCHAR(NNN)) FROM Vertriebsdaten
```

Wenn Sie in diesem Beispiel:

- eine Datei in das Medienfeld mithilfe von FileMaker Pro eingefügt haben, aber nur einen Verweis auf die Datei gespeichert haben, ruft die SELECT-Anweisung die Dateiverweisinformationen als Typ SQL\_VARCHAR ab.
- den Inhalt einer Datei in das Medienfeld mithilfe von FileMaker Pro eingefügt haben, ruft die SELECT-Anweisung den Namen der Datei ab.
- eine Datei in das Medienfeld von einer anderen Anwendung importiert haben, zeigt die SELECT-Anweisung '?' an (die Datei wird als Untitled.dat in FileMaker Pro angezeigt).

Um Daten aus einem Medienfeld abzurufen, verwenden Sie die Funktion GetAs und geben Sie den Typ der Datei an, je nachdem, wie die Daten in das Medienfeld in FileMaker Pro eingegeben wurden.

- Wenn die Daten mit dem Befehl Einfügen > Datei eingefügt wurden, geben Sie 'FILE' in der Funktion GetAs an. Beispiel:

```
SELECT GetAs(Firmenprospekte, 'FILE') FROM Vertriebsdaten
```

- Wenn die Daten mit dem Befehl Einfügen > Ton (Standardton – MAC OS-Rohformat) eingefügt wurden, geben Sie 'snd' in der Funktion GetAs an. Beispiel:

```
SELECT GetAs(Firmenmeeting, 'snd') FROM Firma_Newsletter
```

- Wenn die Daten mit dem Befehl Einfügen > Objekt eingefügt wurden, geben Sie 'EMBO' in der Funktion GetAs an. Beispiel:

```
SELECT GetAs(Firmenergebnisse, 'EMBO') FROM Annual_Report
```

- Wenn die Daten mit dem Befehl Einfügen > Bild eingefügt wurden, ziehen und legen Sie ab, fügen Sie aus der Zwischenablage ein, geben Sie einen der Dateitypen (Groß-/Kleinschreibung beachten) ein, die in der folgenden Tabelle genannt werden. Beispiel:

```
SELECT GetAs(Firmenlogo, 'JPEG') FROM Firmensymbole
```

Dateityp	Beschreibung	Dateityp	Beschreibung
'EMF+'	Windows Enhanced Metafile Plus	PDF'	Portable Document Format
'EPS'	Embedded PostScript	PICT'	Mac OS (hat keinen dateibasierten 512-Byte-Header)
'FPix'	Flash (FPX)	'PNGf'	Bitmap-Bildformat
'FORK'	Ressourcenteil (Mac OS)	'PNTG'	MacPaint
'GIFf'	Graphics Interchange Format	qtif'	QuickTime-Bilddatei
'JPEG'	Fotografische Bilder	'.SGI'	Generisches Bitmap-Format
'JP2'	JPEG 2000	'TIFF'	Raster-Dateiformat für digitale Bilder

Dateityp	Beschreibung	Dateityp	Beschreibung
'META'	Windows-Metafile (erweitert)	TPIC'	Targa
'METO'	Windows-Metafile (original)	XMLO'	Layoutobjekte
'moov'	Altes QuickTime-Format (Mac OS)	'8BPS'	PhotoShop (PSD)

## DELETE-Anweisung

Verwenden Sie die DELETE-Anweisung, um Datensätze aus einer Datenbanktabelle zu löschen. Das Format der DELETE-Anweisung ist:

```
DELETE FROM tabellenname [ WHERE { bedingungen } ]
```

**Hinweis** Die WHERE-Klausel legt fest, welche Datensätze gelöscht werden. Wenn Sie das Schlüsselwort WHERE nicht verwenden, werden alle Datensätze in der Tabelle gelöscht (aber die Tabelle bleibt intakt).

Ein Beispiel einer DELETE-Anweisung in der Angestellten-Tabelle lautet:

```
DELETE FROM ang WHERE angnr = 'A10001'
```

Jede DELETE-Anweisung entfernt jeden Datensatz, der die Bedingungen der WHERE-Klausel erfüllt. In diesem Fall wird jeder Datensatz gelöscht, der die Angestelltennummer A10001 hat. Da in der Angestellten-Tabelle Angestelltennummern eindeutig sind, wird nur ein Datensatz gelöscht.

## INSERT-Anweisung

Verwenden Sie die INSERT-Anweisung, um Datensätze in einer Datenbanktabelle zu erstellen. Sie können angeben:

- eine Liste von Werten, die als neuer Datensatz eingefügt werden
- eine SELECT-Anweisung, die als neuen Satz von Datensätzen einzufügende Daten aus einer anderen Tabelle kopiert

Das Format der INSERT-Anweisung ist:

```
INSERT INTO tabellenname [(spaltenname, ...)] VALUES (ausdr, ...)
```

*spaltenname* ist eine optionale Liste von Spaltennamen, die den Namen und die Reihenfolge der Spalten angibt, deren Werte in der VALUES-Klausel angegeben sind. Wenn Sie *spaltenname* nicht angeben, müssen die Wertausdrücke (*ausdr*) Werte für alle in der Tabelle definierten Spalten angeben und in der gleichen Reihenfolge sein wie die für die Tabelle definierten Spalten.

*ausdr* ist die Liste der Ausdrücke, die Werte für die Spalten des neuen Datensatzes zur Verfügung stellt. Gewöhnlich sind die Ausdrücke konstante Werte für die Spalten (sie können aber auch Unterabfragen sein). Sie müssen Zeichenfolgenwerte in einzelnen Anführungszeichen (') angeben. Um ein einfaches Anführungszeichen in einer Zeichenfolge, die durch einfache Anführungszeichen eingeschlossen ist, aufzunehmen, verwenden Sie zwei einfache Anführungszeichen (z. B. ist"s). Datums-, Zeit- und Zeitstempelwerte müssen von geschweiften Klammern {} umgeben sein. Logische Werte, die Zeichen sind, müssen von Punkten umgeben sein (z. B. .T. oder .F.). Unterabfragen müssen in Klammern angegeben werden.

Das folgende Beispiel fügt eine Liste von Ausdrücken ein:

```
INSERT INTO ang (nachname, vorname, angnr, gehalt, einstdatum)
VALUES ('Schmidt', 'Johann', 'A22345', 27500, {6.5.2004})
```

Jede INSERT-Anweisung fügt der Datenbanktabelle einen Datensatz hinzu. In diesem Fall wurde der Angestellten-Datenbanktabelle ANG ein Datensatz hinzugefügt. Werte werden für fünf Spalten angegeben. Den restlichen Spalten in der Tabelle wird ein leerer Wert, also null, zugeordnet.

**Hinweis** In Medienfelder können Sie nur Text einfügen (INSERT).

Die SELECT-Anweisung ist eine Abfrage, die Werte für jeden in der Spaltenname-Liste angegebenen Wert spaltenname zurückgibt. Die Verwendung einer SELECT-Anweisung anstelle einer Liste von Wertausdrücken ermöglicht Ihnen, eine Menge von Zeilen aus einer Tabelle auszuwählen und sie in eine andere Tabelle mit einer einzelnen INSERT-Anweisung einzufügen.

Hier ein Beispiel einer INSERT-Anweisung, die eine SELECT-Anweisung verwendet:

```
INSERT INTO ang1 (vorname, nachname, angnr, abt, gehalt)
SELECT vorname, nachname, angnr, abt, gehalt FROM ang
WHERE abt = 'D050'
```

In dieser Art von INSERT-Anweisung muss die Anzahl der einzufügenden Spalten der Anzahl der Spalten in der SELECT-Anweisung entsprechen. Die Liste der einzufügenden Spalten muss den Spalten in der SELECT-Anweisung so entsprechen, wie sie einer Liste von Wertausdrücken in einer anderen Art von INSERT-Anweisung entsprechen würde. Zum Beispiel entspricht die erste eingefügte Spalte der ersten ausgewählten Spalte, die zweite eingefügte der zweiten usw.

Größe und Datentyp dieser entsprechenden Spalten müssen kompatibel sein. Jede Spalte in der SELECT-Liste sollte über einen Datentyp verfügen, den der ODBC- bzw. JDBC-Treiber bei einem regulären INSERT/UPDATE der entsprechenden Spalte in der INSERT-Liste akzeptiert. Werte werden abgeschnitten, wenn die Größe des Werts in der SELECT-Listenspalte größer als die Größe der entsprechenden INSERT-Listenspalte ist.

Die SELECT-Anweisung wird vor allen eingefügten Werten ausgewertet.

## UPDATE-Anweisung

Verwenden Sie die UPDATE-Anweisung, um Datensätze in einer Datenbanktabelle zu ändern. Das Format der UPDATE-Anweisung ist:

```
UPDATE tabellenname SET spaltenname = ausdr, ... [ WHERE { bedingungen } ]
```

spaltenname ist der Name einer Spalte, deren Wert zu ändern ist. Mehrere Spalten können in einer Anweisung geändert werden.

ausdr ist der neue Wert für die Spalte. Gewöhnlich sind die Ausdrücke konstante Werte für die Spalten (sie können aber auch Unterabfragen sein). Sie müssen Zeichenfolgenwerte in einzelnen Anführungszeichen (') angeben. Um ein einfaches Anführungszeichen in einer Zeichenfolge, die durch einfache Anführungszeichen eingeschlossen ist, aufzunehmen, verwenden Sie zwei einfache Anführungszeichen (z. B. ist"s). Datums-, Zeit- und Zeitstempelwerte müssen von geschweiften Klammern { } umgeben sein. Logische Werte, die Zeichen sind, müssen von Punkten umgeben sein (z. B. .T. oder .F.). Unterabfragen müssen in Klammern angegeben werden.

Die WHERE-Klausel ist jede gültige Klausel. Sie bestimmt, welche Datensätze aktualisiert werden.

Ein Beispiel einer UPDATE-Anweisung in der Angestellten-Tabelle lautet:

```
UPDATE ang SET gehalt=32000, steuerfrei=1 WHERE angnr = 'A10001'
```

Die UPDATE-Anweisung ändert jeden Datensatz, der die Bedingungen der WHERE-Klausel erfüllt. In diesem Fall werden Gehalt und Steuerfreiheit für alle Angestellten mit der Angestelltennummer A10001 geändert. Da in der Angestellten-Tabelle Angestelltennummern eindeutig sind, wird nur ein Datensatz aktualisiert.

Hier sehen Sie ein Beispiel mit einer Unterabfrage:

```
UPDATE ang SET gehalt = (SELECT avg(gehalt) from ang ) WHERE angnr = 'A10001'
```

In diesem Fall wird das Gehalt für jeden Angestellten mit Angestelltennummer A10001 auf den Gehaltsmittelwert des Unternehmens geändert.

**Hinweis** In Medienfeldern können Sie nur Text aktualisieren (UPDATE).

## CREATE TABLE-Anweisung

Verwenden Sie die CREATE TABLE-Anweisung, um eine Tabelle in einer Datenbankdatei zu erstellen. Das Format der CREATE TABLE-Anweisung ist:

```
CREATE TABLE tabellenname tabellenelementliste [NOT NULL]
```

In der Anweisung geben Sie Namen und Datentyp jeder Spalte an.

tabellenname und tabellenelementliste haben eine Beschränkung auf 100 Zeichen. Die Definition einer Spalte als NOT NULL wählt automatisch die Überprüfungsoption Nicht leer für das entsprechende Feld in der FileMaker-Datenbankdatei aus. Das Feld wird als Wert erforderlich im Register Felder des Dialogfelds „Datenbank verwalten“ in FileMaker Pro markiert.

### Beispiele

Verwendung	Beispiel-SQL
Textspalte	CREATE TABLE T1 (C1 VARCHAR, C2 VARCHAR (50), C3 VARCHAR (1001), C4 VARCHAR (500276))
Textspalte, NOT NULL	CREATE TABLE T1NN (C1 VARCHAR NOT NULL, C2 VARCHAR (50) NOT NULL, C3 VARCHAR (1001) NOT NULL, C4 VARCHAR (500276) NOT NULL)
Zahlenspalte	CREATE TABLE T2 (C1 DECIMAL, C2 DECIMAL (10,0), C3 DECIMAL (7539,2), C4 DECIMAL (497925,301))
Datumsspalte	CREATE TABLE T3 (C1 DATE, C2 DATE, C3 DATE, C4 DATE)
Zeitspalte	CREATE TABLE T4 (C1 TIME, C2 TIME, C3 TIME, C4 TIME)
Zeitstempelspalte	CREATE TABLE T5 (C1 TIMESTAMP, C2 TIMESTAMP, C3 TIMESTAMP, C4 TIMESTAMP)
BLOB-Spalte	CREATE TABLE T6 (C1 BLOB, C2 BLOB, C3 BLOB, C4 BLOB)

## ALTER TABLE-Anweisung

Verwenden Sie die ALTER TABLE-Anweisung, um die Struktur einer bestehenden Tabelle in einer Datenbankdatei zu ändern. Sie können in jeder Anweisung nur eine Spalte ändern. Die Formate der ALTER TABLE-Anweisung sind:

```
ALTER TABLE tabellenname ADD [COLUMN] spaltendefinition
```

```
ALTER TABLE tabellenname DROP [COLUMN] unqualifizierter_spaltenname
```

Sie müssen die Struktur der Tabelle kennen und wissen, wie Sie sie ändern, bevor Sie die ALTER TABLE-Anweisung verwenden.

### Beispiele

Für	Beispiel-SQL
Spalten hinzuzufügen	ALTER TABLE Verkaeufer ADD C1 VARCHAR
Spalten zu entfernen	ALTER TABLE Verkaeufer DROP C1

## CREATE INDEX-Anweisung

Verwenden Sie die CREATE INDEX-Anweisung, um Suchen in einer Datenbankdatei zu beschleunigen. Das Format der CREATE INDEX-Anweisung ist:

```
CREATE INDEX [ indexname ] [ON] tabellenname.spaltenname
```

CREATE INDEX wird für eine einzelne Spalte unterstützt (Mehrspaltenindizes werden nicht unterstützt). Indizes sind bei Spalten nicht zulässig, die Medienfeldtypen, Statistikfeldern, Feldern mit globaler Speicherung oder nicht gespeicherten Formelfeldern in einer FileMaker-Datenbankdatei entsprechen.

Das Erstellen eines Index für eine Textspalte wählt die Speicheroption **Minimal** unter Indizierung für das entsprechende Feld in der FileMaker-Datenbankdatei automatisch aus. Das Erstellen eines Index für eine Nicht-Textspalte (oder eine als japanischer Text formatierte Spalte) wählt die Speicheroption **Alle** unter Indizierung für das entsprechende Feld in der FileMaker-Datenbankdatei automatisch aus.

Das Erstellen eines Index für eine beliebige Spalte wählt die Speicheroption **Indizes bei Bedarf** automatisch erstellen unter Indizierung für das entsprechende Feld in der FileMaker-Datenbankdatei automatisch aus.

#### Beispiel:

```
CREATE INDEX meinIndex ON Verkaeufer.Verkaeufern
```

### DROP INDEX-Anweisung

Verwenden Sie die **DROP INDEX**-Anweisung, um einen Index aus einer Datenbankdatei zu entfernen. Das Format der **DROP INDEX**-Anweisung ist:

```
DROP INDEX [ON] tabellenname.spaltenname
```

Entfernen Sie einen Index, wenn Ihre Datenbankdatei zu groß ist oder Sie ein Feld nicht häufig in Abfragen verwenden.

Wenn Ihre Abfragen langsam ausgeführt werden und Sie mit einer sehr großen FileMaker-Datenbankdatei mit vielen indizierten Textfeldern arbeiten, sollten Sie in Erwägung ziehen, die Indizes einiger Felder zu entfernen. Erwägen Sie auch, die Indizes von Feldern zu entfernen, die Sie selten in **SELECT**-Anweisungen verwenden.

Das Entfernen eines Index für eine beliebige Spalte wählt die Speicheroption **Ohne** unter Indizierung für das entsprechende Feld in der FileMaker-Datenbankdatei automatisch aus und deaktiviert die Option **Indizes bei Bedarf** automatisch erstellen.

Das Attribut **PREVENT INDEX CREATION** wird nicht unterstützt.

#### Beispiel:

```
DROP INDEX ON Verkaeufer.Verkaeufern
```

### SQL-Statistikfunktionen:

Statistikfunktionen geben einen Wert aus einer Menge von Datensätzen zurück. Sie können eine Statistikfunktion als Teil einer **SELECT**-Anweisung mit einem Feldnamen (zum Beispiel **AVG(GEHALT)**) oder in Kombination mit einem Spaltenausdruck (zum Beispiel **AVG(GEHALT \* 1,07)**) verwenden.

Sie können dem Spaltenausdruck den **DISTINCT**-Operator voranstellen, um doppelte Werte zu eliminieren. Beispiel:

```
COUNT (DISTINCT nachname)
```

In diesem Beispiel werden nur eindeutige Nachnamenswerte gezählt.

**Wichtig** Verwenden Sie für **SQL**-Funktionsnamen Großbuchstaben (manche unterscheiden Groß-/Kleinschreibung).

Statistikfunktion	Liefert den Wert
SUM	Die Summe der Werte in einem Zahlenfeldausdruck. Beispiel: <b>SUM(GEHALT)</b> gibt die Summe aller Gehaltsfeldwerte zurück.
AVG	Der Mittelwert der Werte in einem Zahlenfeldausdruck. Beispiel: <b>AVG(GEHALT)</b> gibt den Mittelwert aller Gehaltsfeldwerte zurück.

Statistikfunktion	Liefert den Wert
COUNT	Die Anzahl der Werte in einem Feldausdruck. Beispiel: COUNT(NAME) gibt die Anzahl aller Namenswerte zurück. Bei Verwendung von COUNT mit einem Feldnamen gibt COUNT die Anzahl der Feldwerte ungleich null zurück. Ein spezielles Beispiel ist COUNT(*), das die Anzahl der Datensätze in einer Menge zurückgibt, einschließlich der Datensätze mit null Werten.
MAX	Der Maximalwert in einem Feldausdruck. Beispiel: MAX(GEHALT) gibt den maximalen Gehaltsfeldwert zurück.
MIN	Der Minimalwert in einem Feldausdruck. Beispiel: MIN(GEHALT) gibt den minimalen Gehaltsfeldwert zurück.

## Beispiele

```
SELECT SUM (Verkaufsdaten.Betrag) AS agg FROM Verkaufsdaten
SELECT AVG (Verkaufsdaten.Betrag) AS agg FROM Verkaufsdaten
SELECT COUNT (Verkaufsdaten.Betrag) AS agg FROM Verkaufsdaten
SELECT MAX (Verkaufsdaten.Betrag) AS agg FROM Verkaufsdaten WHERE
Verkaufsdaten.Betrag < 3000
SELECT MIN (Verkaufsdaten.Betrag) AS agg FROM Verkaufsdaten WHERE
Verkaufsdaten.Betrag < 3000
```

## SQL-Ausdrücke

Verwenden Sie in den Klauseln WHERE, HAVING und ORDER BY SELECT-Anweisungen, um detaillierte und raffinierte Datenbankabfragen zu erstellen. Gültige Ausdruckelemente sind:

Feldnamen	Numerische Operatoren	Relationale Operatoren
Konstanten und Literale	Zeichenoperatoren	Logische Operatoren
Exponentialschreibweise	Datumsoperatoren	Funktionen

## Feldnamen

Der gängigste Ausdruck ist ein einfacher Feldname wie `formel` oder `Verkaufsdaten.Rechnungsnr.`

## Konstanten und Literale

Konstanten sind Werte, die sich nicht ändern. Zum Beispiel ist im Ausdruck `PREIS * 1,05` der Wert `1,05` eine Konstante. Oder Sie weisen der Konstante `Anzahl_der_Tage_im_Juni` einen Wert von `30` zu.

Ein Literal ist eine andere Art von Konstante, aber anstatt einen zugewiesenen Wert zu haben, ist das Literal selbst der Wert wie `'Paris'` oder `'14:35:10'`. Ein Literal ist eine Konstante nach dem Prinzip „what you see is what you get“.

Sie müssen Zeichenkonstanten (wie Literale) in einzelnen Anführungszeichen (') angeben. Um ein einfaches Anführungszeichen in einer Zeichenkonstanten, die durch einfache Anführungszeichen eingeschlossen ist, aufzunehmen, verwenden Sie zwei einfache Anführungszeichen (z. B. `ist''s`).

Sie müssen Datums-, Zeit- und Zeitstempelkonstanten in geschweiften Klammern ({} ) angeben, zum Beispiel `{D '2005-06-05'}`, `{14:35:10}` oder `{TS '2005-06-05 14:35:10'}`. Ausnahme: SQL-92-Syntax verlangt ISO-Datums- und Zeitformate ohne Klammern:

- DATE 'JJJ-MM-TT'
- TIME 'HH:MM:SS'
- TIMESTAMP 'JJJ-MM-TT HH:MM:SS'

Konstante	Akzeptable Syntax (Beispiele)
Text	'Paris'
Zahl	1.05
Datum	DATE '2005-06-05' { D '2005-06-05' } { 06/05/2005 } { 06/05/05 }
Zeit	TIME '14:35:10' { T '14:35:10' } { 14:35:10 }
Zeitstempel	TIMESTAMP '2005-06-05 14:35:10' { TS '2005-06-05 14:35:10' } { 06/05/2005 14:35:10 } { 2005-06-05 14:35:10 } { 06/05/05 14:35:10 } Stellen Sie sicher, dass Strenger Datentyp: Vierstellige Jahreszahl nicht als Überprüfungsoption in der FileMaker-Datenbankdatei für ein Feld ausgewählt ist, das diese zweistellige Jahressyntax verwendet.

Wenn Sie Datums- und Zeitwerte eingeben, verwenden Sie das Format der Sprachumgebung der Datenbankdatei. Wenn die Datenbank z. B. in einem italienischen Sprachsystem erstellt wurde, verwenden Sie die italienischen Datums- und Zeitformate.

Logische Werte, die aus Zeichen bestehen, müssen zwischen Punkten angegeben werden. Die logischen Konstanten sind .T. und 1 für Wahr und .F. und 0 für Falsch. Verwenden Sie 1 und 0 aus Portabilitätsgründen.

## Exponentialschreibweise

Sie können Exponentialschreibweise verwenden:

### Beispiel:

```
SELECT spalte1, 3.4E+7 FROM tabelle1 WHERE formel < 3.4E-6 * spalte2
```

## Numerische Operatoren

In Zahlenausdrücken können Sie folgende Operatoren aufnehmen: +, -, \*, /, und ^ oder \*\* (Exponent). Sie können numerischen Ausdrücken ein Plus (+) oder Minus (-) voranstellen.

## Zeichenoperatoren

Sie können Zeichen verketteten.

### Beispiele

In den folgenden Beispielen ist nachname 'JONAS ' und vorname 'ROBERT ':

Operator	Verkettung	Beispiel:	Ergebnis
+	Führende Leerzeichen beibehalten	vorname + nachname	'ROBERT JONAS '
-	Führende Leerzeichen ans Ende bewegen	vorname - nachname	'ROBERTJONAS '

## Datumsoperatoren

Sie können Datumswerte verändern.

### Beispiele

In den folgenden Beispielen ist einstdatum {01/30/2004}.

Operator	Wirkung auf Datum	Beispiel:	Ergebnis
+	Einem Datum eine Anzahl von Tagen hinzufügen	einstdatum + 5	{02/04/2004}
-	Die Anzahl von Tagen zwischen zwei Datumswerten ermitteln oder eine Anzahl von Tagen von einem Datum abziehen	einstdatum - {01/01/2004} einstdatum - 10	29 {01/20/2004}

Weitere Beispiele:

```
SELECT Verkaufsdatum, Verkaufsdatum + 30 AS agg FROM Verkaufsdaten
SELECT Verkaufsdatum, Verkaufsdatum - 30 AS agg FROM Verkaufsdaten
```

## Relationale Operatoren

Operator	Bedeutung
=	Ist gleich
<>	Ist ungleich
>	Größer als
>=	Größer oder gleich
<	Kleiner als
<=	Kleiner oder gleich
LIKE	Entspricht einem Muster
NOT LIKE	Entspricht nicht einem Muster
IS NULL	Ist gleich null
IS NOT NULL	Ist nicht gleich null
BETWEEN	Bereich von Werten zwischen einer unteren und oberen Grenze
IN	Teil einer Menge von angegebenen Werten oder Teil einer Unterabfrage
NOT IN	Nicht Teil einer Menge von angegebenen Werten oder Teil einer Unterabfrage
EXISTS	„Wahr“, wenn eine Unterabfrage wenigstens einen Datensatz zurückgibt.
ANY	Vergleicht einen Wert mit jedem Wert, der von einer Unterabfrage zurückgegeben wird (dem Operator muss ein =, <>, >, >=, < oder <= vorangestellt sein); =Any entspricht _In_
ALL	Vergleicht einen Wert mit jedem Wert, der von einer Unterabfrage zurückgegeben wird (dem Operator muss ein =, <>, >, >=, < oder <= vorangestellt sein).

### Beispiele

```
SELECT Vertriebsdaten.Rechnungsnr FROM Vertriebsdaten
WHERE Verkaufsdaten.Verkaeuferrnr = 'SP-1'
SELECT Verkaufsdaten.Betrag FROM Verkaufsdaten WHERE
Verkaufsdaten.Rechnungsnr <> 125
SELECT Verkaufsdaten.Betrag FROM Verkaufsdaten WHERE Verkaufsdaten.Betrag
> 3000
```

```

SELECT Verkaufsdaten.Verkaufszeit FROM Verkaufsdaten
WHERE Verkaufsdaten.Verkaufszeit < '12:00:00'
SELECT Vertriebsdaten.Firmenname FROM Vertriebsdaten
WHERE Vertriebsdaten.Firmenname LIKE '%Universität'
SELECT Vertriebsdaten.Firmenname FROM Vertriebsdaten
WHERE Vertriebsdaten.Firmenname NOT LIKE '%Universität'
SELECT Vertriebsdaten.Betrag FROM Vertriebsdaten WHERE Vertriebsdaten.Betrag
IS NULL
SELECT Vertriebsdaten.Betrag FROM Vertriebsdaten WHERE Vertriebsdaten.Betrag
IS NOT NULL
SELECT Vertriebsdaten.Rechnungsnr FROM Vertriebsdaten
WHERE Vertriebsdaten.Rechnungsnr BETWEEN 1 AND 10
SELECT COUNT (Verkaufsdaten.Rechnungsnr) AS agg
FROM Verkaufsdaten WHERE Verkaufsdaten.Rechnungsnr IN (50,250,100)
SELECT COUNT (Verkaufsdaten.Rechnungsnr) AS agg
FROM Verkaufsdaten WHERE Verkaufsdaten.Rechnungsnr NOT IN (50,250,100)
SELECT COUNT (Verkaufsdaten.Rechnungsnr) AS agg FROM Verkaufsdaten
WHERE Verkaufsdaten.Rechnungsnr NOT IN (SELECT Verkaufsdaten.Rechnungsnr
FROM Verkaufsdaten WHERE Verkaufsdaten.Verkaeufern = 'SP-4')
SELECT *
FROM Verkaufsdaten WHERE EXISTS (SELECT Verkaufsdaten.Betrag
FROM Verkaufsdaten WHERE Verkaufsdaten.Verkaeufern IS NOT NULL)
SELECT *
FROM Verkaufsdaten WHERE Verkaufsdaten.Betrag = ANY (SELECT
Verkaufsdaten.Betrag
FROM Verkaufsdaten WHERE Verkaufsdaten.Verkaeufern = 'SP-1')
SELECT *
FROM Verkaufsdaten WHERE Verkaufsdaten.Betrag = ALL (SELECT
Verkaufsdaten.Betrag
FROM Verkaufsdaten WHERE Verkaufsdaten.Verkaeufern IS NULL)

```

## Logische Operatoren

Sie können zwei oder mehrere Bedingungen kombinieren. Die Bedingungen müssen mit AND oder OR in Beziehung stehen:

```
gehalt = 40000 AND steuerfrei = 1
```

Der logische Operator NOT wird verwendet, um die Bedeutung umzukehren:

```
NOT (gehalt = 40000 AND steuerfrei = 1)
```

## Beispiele

```

SELECT * FROM Vertriebsdaten WHERE Vertriebsdaten.Firmenname
NOT LIKE '%Universität' AND Vertriebsdaten.Betrag > 3000
SELECT * FROM Vertriebsdaten WHERE (Vertriebsdaten.Firmenname
LIKE '%Universität' OR Vertriebsdaten.Betrag > 3000)
AND Verkaufsdaten.Verkaeufern = 'SP-1'

```

## Funktionen

ODBC- und JDBC-Client-Treiber unterstützen viele Funktionen, die Sie in Ausdrücken verwenden können. Einige der Funktionen geben Zeichenfolgen zurück, einige Zahlen und einige Datumswerte.

**Wichtig** Verwenden Sie für SQL-Funktionsnamen Großbuchstaben (manche unterscheiden Groß-/Kleinschreibung).

### Funktionen, die Zeichenfolgen zurückgeben

Funktionen, die Zeichenfolgen zurückgeben	Beschreibung	Beispiel:
CHR	Konvertiert einen ASCII-Code in eine Zeichenfolge mit einem Zeichen	CHR(67) ergibt C
RTRIM	Entfernt nachfolgende Leerzeichen aus einer Zeichenfolge	RTRIM('ABC ') ergibt ABC.
TRIM	Entfernt führende und nachfolgende Leerzeichen aus einer Zeichenfolge	TRIM(' ABC ') ergibt ABC.
LTRIM	Entfernt führende Leerzeichen aus einer Zeichenfolge	LTRIM(' ABC') ergibt ABC.
UPPER	Ändert jeden Buchstaben einer Zeichenfolge zu Großbuchstaben	UPPER('Allen') ergibt ALLEN.
LOWER	Ändert jeden Buchstaben einer Zeichenfolge zu Kleinbuchstaben	LOWER('Allen') ergibt allen.
LEFT	Gibt die Zeichen von links einer Zeichenfolge zurück	LEFT('Mattson', 3) ergibt Mat.
RIGHT	Gibt die Zeichen von rechts einer Zeichenfolge zurück	RIGHT('Mattson', 4) ergibt tson.
SUBSTR	Gibt einer Unterzeichenfolge einer Zeichenfolge mit Parametern der Zeichenfolge, dem ersten zu extrahierenden Zeichen und der Anzahl der zu extrahierenden Zeichen (optional) zurück.	SUBSTR('Konrad', 2, 3) ergibt onr. SUBSTR('Konrad', 2) ergibt onrad.
SPACE	Erzeugt eine Zeichenfolge mit Leerzeichen	SPACE(5) ergibt ' '
STRVAL	Konvertiert einen Wert beliebigen Typs in eine Buchstabenzeichenfolge	STRVAL('Woltman') ergibt Woltman STRVAL(5 * 3) ergibt 15 STRVAL(4 = 5) ergibt 'False' STRVAL({12/25/2004}) ergibt 12/25/2004
TIME	Ergibt die Tageszeit als Zeichenfolge	Um 21:49 ergibt TIME() 21:49:00
USERNAME	Gibt die zur Verbindungszeit angegebene Anmeldekennung zurück.	

### Beispiele

```
SELECT CHR(67) + SPACE(1) + CHR(70) FROM Verkaeufer
SELECT RTRIM(' ' + Verkaeufer.Verkaeuferrn) AS agg FROM Verkaeufer
SELECT TRIM(SPACE(1) + Verkaeufer.Verkaeuferrn) AS agg FROM Verkaeufer
SELECT LTRIM(' ' + Verkaeufer.Verkaeuferrn) AS agg FROM Verkaeufer
SELECT UPPER(Verkaeufer.Verkaeuferrn) AS agg FROM Verkaeufer
SELECT LOWER(Verkaeufer.Verkaeuferrn) AS agg FROM Verkaeufer
SELECT LEFT(Verkaeufer.Verkaeuferrn, 5) AS agg FROM Verkaeufer
SELECT RIGHT(Verkaeufer.Verkaeuferrn, 7) AS agg FROM Verkaeufer
SELECT SUBSTR(Verkaeufer.Verkaeuferrn, 2, 2) +
SUBSTR(Verkaeufer.Verkaeuferrn, 4, 2) AS agg FROM Verkaeufer
```

```

SELECT SUBSTR(Verkaeuf.er.Verkaeuf ernr, 2) + SUBSTR(Verkaeuf.er.Verkaeuf ernr,
4) AS agg FROM Verkaeuf.er
SELECT SPACE(2) + Verkaeuf.er.Verkaeuf ernr AS Verkaeuf ernr FROM Verkaeuf.er
SELECT STRVAL('60506') AS agg FROM Verkaufsdaten WHERE
Verkaufsdaten.Rechnungsnr = 1
SELECT TIME() AS agg FROM Verkaufsdaten WHERE Verkaufsdaten.Rechnungsnr = 1
SELECT USERNAME() AS agg FROM Verkaufsdaten WHERE Verkaufsdaten.Rechnungsnr = 1
    
```

**Funktionen, die Zahlen zurückgeben**

<b>Funktionen, die Zahlen zurückgeben</b>	<b>Beschreibung</b>	<b>Beispiel:</b>
MOD	Teilt zwei Zahlen und gibt den Rest der Division zurück.	MOD(10, 3) ergibt 1
LEN	Gibt die Länge einer Zeichenfolge zurück.	LEN('ABC') ergibt 3
MONTH	Gibt den Monatsteil eines Datums zurück.	MONTH({01/30/2004}) ergibt 1
DAY	Gibt den Tagesteil eines Datums zurück.	DAY({01/30/2004}) ergibt 30
YEAR	Gibt den Jahressteil eines Datums zurück.	YEAR({01/30/2004}) ergibt 2004
MAX	Gibt die größere von zwei Zahlen zurück.	MAX(66, 89) ergibt 89
DAYOFWEEK	Gibt den Tag der Woche (1-7) eines Datumsausdrucks zurück.	DAYOFWEEK({05/01/2004}) ergibt 7
MIN	Gibt die kleinere von zwei Zahlen zurück.	MIN(66, 89) ergibt 66
POW	Erhebt eine Zahl zu einer Potenz.	POW(7, 2) ergibt 49
INT	Gibt den ganzzahligen Teil einer Zahl zurück.	INT(6, 4321) ergibt 6
X	Gibt die dezimale Entsprechung einer Hexadezimalzahl zurück.	X'b9' ergibt 185
B	Gibt die dezimale Entsprechung einer Binärzahl zurück.	B'1001' ergibt 9
ROUND	Rundet eine Zahl.	ROUND(123.456, 0) ergibt 123 ROUND(123.456, 2) ergibt 123.46 ROUND(123.456, -2) ergibt 100
NUMVAL	Konvertiert eine Buchstabenzeichenfolge in eine Zahl; wenn die Buchstabenzeichenfolge keine gültige Zahl ist, wird 0 zurückgegeben.	NUMVAL('123') ergibt 123
VAL	Konvertiert eine Buchstabenzeichenfolge in eine Zahl; wenn die Buchstabenzeichenfolge keine gültige Zahl ist, wird 0 zurückgegeben.	VAL('123') ergibt 123

**Funktionen, die Datumswerte zurückgeben**

<b>Funktionen, die Datumswerte zurückgeben</b>	<b>Beschreibung</b>	<b>Beispiel:</b>
DATE	Gibt das aktuelle Systemdatum zurück.	Wenn heute der 21.11.2005 ist, ergibt DATE() {2005-11-21}.
DATEVAL	Konvertiert eine Buchstabenzeichenfolge in ein Datum.	DATEVAL('01/30/2006') ergibt {2006-01-30}.

## Priorität der Operatoren

Wenn die Ausdrücke komplexer werden, wird die Reihenfolge wichtig, in der die Ausdrücke ausgewertet werden. Diese Tabelle zeigt die Reihenfolge, in der die Operatoren ausgewertet werden. Die Operatoren in der ersten Zeile werden zuerst ausgewertet usw. Operatoren in der gleichen Zeile werden im Ausdruck von links nach rechts ausgewertet.

Priorität	Operator
1	Vorzeichen '-', Vorzeichen '+'
2	^, **
3	*, /
4	+, -
5	=, <>, <, <=, >, >=, Like, Not Like, Is Null, Is Not Null, Between, In, Exists, Any, All
6	Hinweis
7	AND
8	ODER

Das folgende Beispiel verdeutlicht die Bedeutung der Priorität:

```
WHERE gehalt > 40000 OR einstdatum > {01/30/2004} UND abt = 'A101'
```

Weil AND zuerst ausgewertet wird, ruft diese Abfrage Angestellte in Abteilung A101 ab, die nach dem 30. Januar 2004 eingestellt wurden, sowie jeden Angestellten mit mehr als 40.000 Euro Gehalt, unabhängig von Abteilung oder Einstellungsdatum.

Um die Klausel in einer anderen Reihenfolge auszuwerten, verwenden Sie Klammern um die zuerst auszuwertenden Bedingungen. Beispiel:

```
WHERE (gehalt > 40000 OR einstdatum > {01/30/2004}) AND abt = 'A101'
```

ruft Angestellte in Abteilung A101 ab, die entweder mehr als 40.000 Euro verdienen oder nach dem 30. Januar 2004 eingestellt wurden.

## ODBC-Katalogfunktionen

Der ODBC-Client-Treiber unterstützt die folgenden Katalogfunktionen:

- SQLTables - Kataloginformationen werden gespeichert und als einzelne Bereichsnamen gemeldet (nur Tabellename).
- SQLColumns
- SQLColumnPrivileges
- SQLDescribeCol
- SQLGetTypeInfo

## JDBC-Metadaten-Funktionen

Der JDBC-Client-Treiber unterstützt die folgenden Metadaten-Funktionen:

- getColumnns
- getColumnPrivileges
- getMetaData
- getTypeInfo
- getTables
- getTableTypes

## Reservierte SQL-Schlüsselwörter

In der folgenden Tabelle werden die reservierten Schlüsselwörter aufgeführt, die nicht als Namen für Spalten, Tabellen, Aliasse oder andere benutzerdefinierte Objekte verwendet werden dürfen. Syntaxfehler können auf die Verwendung dieser reservierten Schlüsselwörter zurückzuführen sein. Wenn Sie eines dieser Schlüsselwörter verwenden möchten, müssen Sie Anführungszeichen einsetzen, um zu vermeiden, dass dieses Wort als Schlüsselwort behandelt wird.

Die folgende Anweisung CREATE TABLE zeigt die Verwendung des Schlüsselworts „OID“ als Datenelementname.

```
create table t ("oid" numeric)
```

### Reservierte Schlüsselwörter

ABSOLUTE	CHARACTER_LEN	DATE	EXTRACT	INTERVAL
ACTION	GTH	DAY	FALSE	INTO
HINZUFÜGEN	CHAR_LENGTH	DAYOFMONTH	FETCH	IS
AFTER	CHECK	DAYOFWEEK	FIRST	ISOLATION
ALIAS	CHR	DEALLOCATE	FLOAT	JOIN
ALL	CLOSE	DEC	FLOOR	KEY
ALLOCATE	COALESCE	DECIMAL	FOR	LANGUAGE
ALTER	COLLATE	DECLARE	FOREIGN	LAST
AND	COLLATION	DEFAULT	FOUND	LCASE
ANY	COLUMN	DEFERRABLE	FROM	LEADING
ARE	COLUMNS	DEFERRED	FULL	LEAVE
AS	COMMIT	DELETE	GENERAL	LEFT
ASC	COMPLETION	DEPTH	GET	LEN
ASSERTION	CONCAT	DESC	GLOBAL	LENGTH
ASYNC	CONNECT	DESCRIBE	GO	LESS
AT	CONNECTION	DESCRIPTOR	GOTO	LEVEL
AUTHORIZATION	CONSTRAINT	DIAGNOSTICS	GRANT	LIKE
AVG	CONSTRAINTS	DICTIONARY	GROUP	LIMIT
BEFORE	CONTINUE	DISCONNECT	HAVING	LOCAL
BEGIN	CONVERT	DISTINCT	HOURL	LOOP
BETWEEN	CORRESPONDING	DOMAIN	IDENTIFY	LOWER
BIT	COUNT	DOUBLE	IF	LTRIM
BIT_LENGTH	CREATE	DROP	IFNULL	MATCH
BOOLEAN	CROSS	EACH	IGNORE	MAX
BOTH	CURDATE	ELSE	IMMEDIATE	MIN
BREADTH	CURRENT	ELSEIF	IN	MINUTE
BY	CURRENT_DATE	END	INDEX	MOD
CALL	CURRENT_TIME	END_EXEC	INDICATOR	MODIFY
CASCADE	CURRENT_TIMEST	EQUALS	INITIALLY	MODULE
CASCADEDED	AMP	ESCAPE	INNER	MONTH
CASE	CURRENT_USER	EXCEPT	INPUT	NAMES
CAST	CURSOR	EXCEPTION	INSENSITIVE	NATIONAL
CATALOG	CURTIME	EXEC	INSERT	NATURAL
CHAR	CYCLE	EXECUTE	INT	NCHAR
CHARACTER	DATA	EXISTS	INTEGER	NEW
		EXTERNAL	INTERSECT	

**Reservierte Schlüsselwörter**

Next	READ	SQLWARNING	WAIT
NO	REAL	STATISTICS	WHEN
NONE	RECURSIVE	STRUCTURE	WHENEVER
NOT	REF	SUBSTR	WHERE
NOW	REFERENCES	SUBSTRING	WHILE
NULL	REFERENCING	SUM	WITH
NULLIF	RELATIVE	SYSTEM_USER	WITHOUT
NUMERIC	ENTFERNEN	TABLE	WORK
OBJECT	REPLACE	TEMPORARY	WRITE
OCTET_LENGTH	RESIGNAL	TEST	YEAR
OF	RESTRICT	THEN	ZONE
OFF	RETURN	THERE	
OID	RETURNS	TIME	
OLD	REVOKE	TIMESTAMP	
ON	RIGHT	TIMEZONE_HOUR	
ONLY	ROLE	TIMEZONE_MINUT	
OPEN	ROLLBACK	E	
OPERATION	ROUND	TO	
OPERATORS	ROUTINE	TRAILING	
OPTION	ROW	TRANSACTION	
ODER	ROW	TRANSLATE	
ORDER	RTRIM	TRANSLATION	
OTHERS	SAVEPOINT	TRIGGER	
OUTER	SCHEMA	TRIM	
OUTPUT	SCROLL	TRUE	
OVERLAPS	SEARCH	TYPE	
PAD	SECOND	UCASE	
PARAMETERS	SECTION	UNDER	
PARTIAL	SELECT	UNION	
PENDANT	SENSITIVE	UNIQUE	
POSITION	SEQUENCE	UNKNOWN	
POW	SESSION	UPDATE	
POWER	SESSION_USER	UPPER	
PRECISION	SET	USAGE	
PREORDER	SIGNAL	USER	
PREPARE	SIMILAR	USERNAME	
PRESERVE	SIZE	USING	
PRIMARY	SMALLINT	VALUE	
PRIOR	SOME	VALUES	
PRIVATE	SPACE	VARCHAR	
PRIVILEGES	SQL	VARIABLE	
PROCEDURE	SQLCODE	VARYING	
PROTECTED	SQLERROR	VIEW	
PUBLIC	SQLEXCEPTION	VIRTUAL	
RCASE	SQLSTATE	VISIBLE	

# Kapitel 6

## Zuordnen von FileMaker-Feldern zu ODBC-Datentypen

Diese Tabelle veranschaulicht, wie FileMaker-Feldtypen den standardmäßigen ODBC-Datentypen zugeordnet werden.

FileMaker-Feldtyp	ODBC-Datentyp	Datentypinformation
Text	SQL_VARCHAR	Die maximale Spaltenlänge für Text beträgt 1 Million Zeichen, außer Sie legen für das Textfeld in FileMaker eine kleinere Maximale Anzahl von Zeichen fest. FileMaker gibt leere Zeichenfolgen als NULL aus.
Zahl	SQL_DOUBLE	Der FileMaker-Zahlenfeld-Typ kann positive oder negative Werte von $10^{-308}$ bis $10^{+308}$ mit bis zu 15 signifikanten Stellen enthalten.
Datum	SQL_DATE	
Zeit	SQL_TIME	Der FileMaker-Feldtyp „Zeit“ kann die Tageszeit oder ein Zeitintervall enthalten. Ein Zeitintervall wird als Tageszeit ausgegeben, außer es ist kleiner als 0 oder größer als 24 Stunden (in beiden Fällen wird der Wert 0 ausgegeben).
Zeitstempel	SQL_TIMESTAMP	
Medienfeld (BLOB)	SQL_LONGVARBINARY	Sie können Binärdaten, Dateiverweisinformationen oder Daten eines angegebenen Dateityps von einem Medienfeld abrufen. Verwenden Sie innerhalb einer SELECT-Anweisung die CAST-Funktion, um die Dateiverweisinformation abzurufen, und verwenden Sie die GetAs-Funktion, um Daten eines bestimmten Dateityps abzurufen.
Formel		Das Ergebnis wird dem entsprechenden ODBC-Datentyp zugeordnet.

Die Länge von Zeichenfolgen in Tabellendeklarationen ist optional. Alle Zeichenfolgen werden in Unicode gespeichert und abgerufen.

### Hinweise

- Sie können bis zu 170 Felder einer FileMaker-Datenbankdatei gleichzeitig mittels SELECT-Anweisung auswählen. Bis zu 100 Felder können Sie gleichzeitig aktualisieren (UPDATE).
- FileMaker unterstützt Wiederholfelder (Array-Datentypen), ODBC hingegen nicht. FileMaker exportiert Wiederholungen in Tab-getrennte oder kommagetrennte Dateien und trennt jede Wiederholung mit einem Gruppentrennzeichen (Unicode-Dezimalwert 29). Textspalten, die mit dem Gruppentrennzeichen getrennt sind, werden verkettet. Alle anderen Datentypen geben nur die erste Wiederholung aus.



# Kapitel 7

## Zuordnen von FileMaker-Feldern zu JDBC-Datentypen

Der JDBC-Client verwendet folgende Zuordnungen, wenn FileMaker-Datentypen in JDBC SQL-Typen konvertiert werden. (Weitere Informationen über diese Datentypen finden Sie auf den Webseiten mit der JDK 1,5-Dokumentation unter [www.javasoft.com](http://www.javasoft.com).)

<b>FileMaker-Feldtyp</b>	<b>JDBC SQL-Typ</b>
Text	java.sql.Types.VARCHAR
Zahl	java.sql.Types.DOUBLE
Datum	java.sql.Types.DATE
Zeit	java.sql.Types.TIME
Zeitstempel	java.sql.Types.TIMESTAMP
Medien	java.sql.Types.BLOB
Formel	angegeben durch den Datentyp des Formelergebnisses

Der JDBC-Client-Treiber konvertiert den FileMaker-Datentyp Formel in den JDBC SQL-Typ, der dem Ergebnis der Formel entspricht. Beispielsweise konvertiert der JDBC-Client-Treiber das Ergebnis einer FileMaker-Formel vom Datentyp Zeitstempel in `java.sql.Types.TIMESTAMP`.



# Kapitel 8

## **ODBC- und JDBC-Fehlermeldungen**

Im Folgenden sind die grundlegenden Fehlermeldungen aufgeführt, die beim Arbeiten mit FileMaker und ODBC/JDBC auftreten können. Eine Auflistung der Fehlernummern und Erläuterungen zu den Fehlern finden Sie unter [www.datadirect.com](http://www.datadirect.com).

Zusätzliche Informationen über den Umgang mit Fehlern in FileMaker erhalten Sie in den Beschreibungen der Funktionen „Hole (Letzte FehlerNr)“ und „Hole (Letzten ODBC Fehler)“ in der FileMaker Pro Hilfe.

### **ODBC-Fehlermeldungen**

Fehlermeldungen werden ausgegeben wegen:

- ODBC-Treiberfehlern
- ODBC-Treiber-Manager-Fehlern
- SequeLink Client-Fehlern
- SequeLink Server-Fehlern
- der Datenquelle oder dem Datenbankverwaltungssystem

### **ODBC-Treiber-Fehlermeldungen**

Ein durch den SequeLink ODBC-Treiber gemeldeter Fehler hat das folgende Format:

[DataDirect] [ODBC SequeLink driver] Meldung

Beispiel:

[DataDirect] [ODBC SequeLink driver] Angegebene Genauigkeit ist ungültig

Falls diese Art von Fehler angezeigt wird, prüfen Sie den letzten ODBC-Aufruf Ihrer Anwendung auf mögliche Probleme oder wenden Sie sich an den Lieferanten Ihrer ODBC-Anwendung.

### **ODBC-Treiber-Manager-Fehlermeldungen**

Ein durch den ODBC-Treiber-Manager gemeldeter Fehler hat das folgende Format:

[Microsoft] [ODBC Driver Manager] Meldung

Beispiel:

[Microsoft] [ODBC Driver Manager] Funktionssequenzfehler

Falls diese Art von Fehler angezeigt wird, prüfen Sie, ob Sie über die richtigen ODBC-Support-Dateien und -Treiber verfügen.

### **SequeLink Client-Fehlermeldungen**

Ein durch den SequeLink ODBC-Client gemeldeter Fehler hat das folgende Format:

[DataDirect] [ODBC SequeLink driver] [SequeLink Client] Meldung

Beispiel:

[DataDirect] [ODBC SequeLink driver] [SequeLink Client] Das angegebene Transkriptionsmodul konnte nicht gefunden werden.

## SequeLink Server-Fehlermeldungen

Ein durch den SequeLink Server gemeldeter Fehler hat das folgende Format:

[DataDirect] [ODBC SequeLink driver] [SequeLink Server] Meldung

Beispiel:

[DataDirect] [ODBC SequeLink driver] [SequeLink Server] Für diese schreibgeschützte Verbindung sind nur SELECT-Anweisungen zulässig.

## Datenquellen-Fehlermeldungen

Eine Fehlermeldung für einen in der Datenquelle auftretenden Fehler enthält den Namen der Datenquelle und hat folgendes Format:

[DataDirect] [ODBC SequeLink driver] [data\_source] Meldung

Beispielsweise können Sie von Ihrer FileMaker-Datenquelle folgende Meldung erhalten:

[DataDirect] [ODBC SequeLink driver] [FileMaker] Benutzername/Passwort ist ungültig

Falls diese Art von Fehler angezeigt wird, haben Sie im Datenbanksystem eine unzulässige Aktion ausgeführt. Lesen Sie hierzu in Ihrer FileMaker-Dokumentation nach oder wenden Sie sich an den Datenbankadministrator.

Aufeinander folgende Meldungen, die sich auf Fehler in verschiedenen Spalten beziehen, können manchmal einen falschen Spaltennamen anzeigen.

## JDBC-Fehlermeldungen

SequeLink für JDBC-Treiber meldet Fehler durch Ausgabe von SQLExceptions an die aufrufende Anwendung. Fehlermeldungen werden ausgegeben wegen:

- JDBC-Treiberfehlern
- SequeLink Server-Fehlern
- der Datenquelle oder dem Datenbankverwaltungssystem

## JDBC-Treiber-Fehlermeldungen

Ein durch den JDBC-Treiber gemeldeter Fehler hat das folgende Format:

[DataDirect] [SequeLink JDBC Driver] Meldung

Beispiel:

[DataDirect] [SequeLink JDBC Driver] Zeitlimit wurde überschritten

Falls diese Art von Fehler angezeigt wird, prüfen Sie den letzten JDBC-Aufruf Ihrer Anwendung auf mögliche Probleme oder wenden Sie sich an den Lieferanten Ihrer JDBC-Anwendung.

## SequeLink Server-Fehlermeldungen

Ein durch den SequeLink Server gemeldeter Fehler hat das folgende Format:

[DataDirect] [SequeLink JDBC Driver] [SequeLink] Meldung

Falls keine SequeLink Server-Fehler vorhanden sind, wird Folgendes angezeigt:

[DataDirect] [JDBC SequeLink driver] [SequeLink]

## Datenquellen-Fehlermeldungen

Eine Fehlermeldung für einen in der Datenquelle auftretenden Fehler enthält den Namen der Datenquelle und hat folgendes Format:

[DataDirect] [SequeLink JDBC Driver] [data\_source] Meldung

Beispielsweise können Sie von Ihrer FileMaker-Datenquelle folgende Meldung erhalten:

[DataDirect] [SequeLink JDBC Driver] [FileMaker] Benutzername/Passwort ist ungültig

Falls diese Art von Fehler angezeigt wird, haben Sie im Datenbanksystem eine unzulässige Aktion ausgeführt. Lesen Sie hierzu in Ihrer FileMaker-Dokumentation nach oder wenden Sie sich an den Datenbankadministrator.



# Index

## A

Aliasse mit SELECT-Anweisung 28, 31  
ALL, Operator 39  
ALTER TABLE (SQL-Anweisung) 35  
Anforderungen für die Installation 11  
ANY, Operator 39  
APIs 7  
ARRAY-Datentyp 22  
Ausdrücke in SQL 37  
Ausschnitte 27  
Automatisch erstellte Schlüssel 22

## B

B, Funktion 42  
Bereitstellen einer FileMaker-Datenquelle 7  
BETWEEN, Operator 39  
Bilddateien in Medienfeldern 32  
bitmap-Dateien in Medienfeldern 32  
BLOB-Datentyp  
  JDBC-Beschränkung 22  
  Verwenden in SELECT 31  
  Verwendung in CREATE TABLE 35  
Boolescher Datentyp 22

## C

CAST-Funktion 32, 47  
CHR, Funktion 41  
Client-Anwendung, Verwenden von FileMaker als 7  
Client-Treiber *s.* Treiber  
CLOB-Datentyp 22  
CREATE INDEX (SQL-Anweisung) 35  
CREATE TABLE (SQL-Anweisung) 35  
Cursor  
  In JDBC 22  
  In ODBC 31

## D

DATALINK-Datentyp 22  
DATE, Funktion 42  
Dateien  
  Einrichten des Zugriffs auf 17  
  Organisation auf einem Computer 8  
  Verwenden in Medienfeldern 32  
Dateinamen-Einschränkungen (Mac OS) 20  
Datenbankverbindungen, Anzahl an unterstützten 8

## Datenquelle

ein Datenquellennamen für jede FileMaker-Datenbankdatei 9  
eine gemeinsam verwendete FileMaker-Datenbankdatei deaktivieren 8  
für Zugriff über JDBC konfigurieren 24  
Konfigurieren für Zugriff über ODBC (Mac OS) 20  
Konfigurieren für Zugriff über ODBC (Windows) 17  
Überprüfen des Zugriffs über ODBC (Windows) 19  
Verwenden von FileMaker als 7  
Zugriff über JDBC überprüfen 25

## Datenquellennamen

Einer pro Datei 9  
Erstellen (MacOS) 20  
Erstellen (Windows) 17  
Schlüsselwortwerte für (Mac OS) 20

Datenquellennamen *siehe* Datenquellennamen

## Datentyp

JDBC-Client-Treiber 49  
ODBC-Client-Treiber 47

## Datentypen zuordnen

JDBC-Client-Treiber 49  
ODBC-Client-Treiber 47

## DATEVAL, Funktion 42

Datumsoperatoren in SQL-Ausdrücken 39

## DAY, Funktion 42

## DAYOFWEEK, Funktion 42

## DELETE (SQL-Anweisung) 33

## DISTINCT, Operator 28

## DROP INDEX (SQL-Anweisung) 36

## E

eine gemeinsam verwendete FileMaker-Datenbankdatei deaktivieren 8

## Erweiterte Zugriffsrechte 17

## Escape-Zeichen 20, 24

## EXISTS, Operator 39

## Exponentialschreibweise in SQL-Ausdrücken 38

## F

## Fehlermeldungsformate 51

## Felder

Zuordnen zu JDBC 49

Zuordnung zu ODBC 47

## Feldnamen in SQL-Ausdrücken 37

FileMaker-Client-Treiber s. Treiber  
 FileMaker-Produkte 8  
 FOR UPDATE (SQL-Klausel) 31  
 FROM (SQL-Klausel) 29  
 Funktionen in SQL-Ausdrücken 41

**G**

GetAs-Funktion 32, 47  
 GROUP BY (SQL-Klausel) 29

**H**

Haltbarer Cursor 22  
 HAVING (SQL-Klausel) 30  
 Hole (Letzte FehlerNr), Funktion 51  
 Hole (Letzten ODBC Fehler), Funktion 51

**I**

INNER JOIN 29  
 INSERT (SQL-Anweisung) 33  
 Installationsanforderungen 11  
 Installieren  
   FileMaker JDBC-Client-Treiber 13  
   ODBC-Treiber 12  
 INT, Funktion 42

**J**

JAR-Archivdateien 13  
 Java Development Kit (JDK) 22  
 Java-Version 12  
 JDBC  
   Beschreibung 21  
   Client-Treiber, beschrieben 21  
   Fehlermeldungen 52  
   Überblick der Verwendung 7  
 JDBC SPI 22  
 JDBC-Client-Treiber  
   beim JDBC-Treiber-Manager registrieren 23  
   Datentypen zuordnen 49  
   Installieren 13  
   JDBC URL angeben 22  
   Metadaten-Funktionen 44  
   Treiberklasse und Haupteinstieg 22  
   Unicode-Unterstützung 27  
   Wiederholfelder 27  
   Zugriff überprüfen 25  
 JDBCTest 14  
 Join 29

**K**

Katalogfunktionen für ODBC 43

Konfigurieren einer FileMaker-Datenquelle  
   über JDBC 24  
   Über ODBC (Mac OS) 20  
   Über ODBC (Windows) 17  
 Konstanten in SQL-Ausdrücken 37  
 Konten und Zugriffsrechte 17

**L**

Leere Werte in Spalten 33  
 Leerzeichen 38  
 Leerzeichen in Datenbankname 20, 24  
 LEFT, Funktion 41  
 LEN, Funktion 42  
 LIKE, Operator 39  
 Literale in SQL-Ausdrücken 37  
 Logische Operatoren in SQL-Ausdrücken 40  
 LOWER, Funktion 41  
 LTRIM, Funktion 41

**M**

Mac OS  
   Erstellen eines Datenquellennamens 20  
   Installation des JDBC-Client-Treibers 13  
   Installation des ODBC-Client-Treibers 13  
   JDBC-Client-Treiber 12  
   ODBC-Administrator 13, 20  
   ODBC-Client-Treiber-Anforderungen 11  
 MAX, Funktion 42  
 Medienfeld  
   JDBC-Datentypzuordnung 49  
   Mit INSERT-Anweisung 33  
   Mit SELECT-Anweisung 32  
   Mit UPDATE-Anweisung 34  
   ODBC-Datentypzuordnung 47  
 Metadaten-Funktionen für JDBC 44  
 Microsoft Abfrage-Assistent 8  
 Microsoft Access  
   Client-Anwendung 8  
 MIN, Funktion 42  
 MOD, Funktion 42  
 MONTH, Funktion 42

**N**

Netzwerkanforderungen 9  
 NICHT-Operator 40  
 NOT NULL (SQL-Klausel) 35  
 Nullwert 33, 47  
 Numerische Operatoren in SQL-Ausdrücken 38  
 NUMVAL, Funktion 42

**O**

## ODBC

- Beschreibung 15
- Fehlermeldungen 51
- Standards, Einhaltung 27
- Überblick der Verwendung 7
- Wiederholfelder 47

ODBC-Administrator (Mac OS) 13, 20

## ODBC-Client-Treiber

- Datentypen zuordnen 47
- Installieren 12
- installieren (Mac OS) 13
- Katalogfunktionen 43
- maximale Anzahl an FileMaker-Feldern 47
- Unicode-Unterstützung 27
- Wiederholfelder 27
- Zugriff überprüfen (Windows) 19

ODBC-Datenquellenadministrator (Windows) 18

ODER-Operator 40

Operatorpriorität bei SQL-Ausdrücken 43

ORDER BY (SQL-Klausel) 30

OUTER JOIN 29

**P**

## Passwort

- Mit JDBC 24
- Mit ODBC 19

## Port

- Angeben für JDBC 24
- Angeben für ODBC (Mac OS) 20
- Angeben für ODBC (Windows) 18

Positioned Updates und Deletes 31

POW, Funktion 42

**Q**

QuickTime-Dateien in Medienfeldern 32

**R**

Rapid Application Development (RAD)-Tools 21

REF-Datentyp 22

Registrieren des JDBC-Client-Treibers 23

Relationale Operatoren in SQL-Ausdrücken 39

Remote-Zugriff 8

Reservierte SQL-Schlüsselwörter 45

RIGHT, Funktion 41

ROUND, Funktion 42

RTRIM, Funktion 41

**S**

## Schlüsselwörter

- für Datenquellennamen (Mac OS) 20
- Reserviert, SQL 45

Secure Socket Layer-Verschlüsselung 25

SELECT (SQL-Anweisung) 27

SequeLink Server Host 18

SequeLink Server Port 18

Server-Datenquelle 18

Sharing, Einrichten von ODBC/JDBC- 17

Sicherungspunkt-Unterstützung 22

SPACE, Funktion 41

Spaltenaliasse 28

Spaltennamen 17

SQL\_C\_BINARY, Datentyp 27

SQL\_C\_WCHAR, Datentyp 27

SQL-92 27

## SQL-Anweisungen

ALTER TABLE 35

CREATE INDEX 35

CREATE TABLE 35

DELETE 33

DROP INDEX 36

INSERT 33

Reservierte Schlüsselwörter 45

SELECT 27

unterstützt von Client-Treibern 27

UPDATE 34

## SQL-Ausdrücke 37

Datumsoperatoren 39

Exponentialschreibweise 38

Feldnamen 37

Funktionen 41

Konstanten 37

Literele 37

logische Operatoren 40

numerische Operatoren 38

Priorität der Operatoren 43

relationale Operatoren 39

Zeichenoperatoren 38

SQLExceptions 52

SQL-Standards, Einhaltung 27

SQL-Statistikfunktionen 36

Standards, Einhaltung 27

Statistikfunktionen in SQL 36

STRVAL, Funktion 41

SUBSTR, Funktion 41

Syntaxfehler 45

Systemanforderungen 11

**T**

Tabellenaliasse 28, 29

Testen des Zugriffs

JDBC-Client-Treiber 25

ODBC-Client-Treiber (Windows) 19

TIME, Funktion 41

Treiber

Deinstallation alter 9

Installation ODBC 12

Installation von FileMaker JDBC 13

Treibereigenschaften

JDBC-Client-Treiber 24

ODBC-Client-Treiber (MacOS) 20

ODBC-Client-Treiber (Windows) 17

TRIM, Funktion 41

## U

Überlegungen zu Datenbanknamen

Mac OS 20

Windows 19

Übersicht

Einrichten von Zugriffsrechten und Sharing 17

Verwenden von ODBC und JDBC mit

FileMaker 7

UND-Operator 40

Unicode-Unterstützung 27

UNION (SQL-Operator) 30

Unterabfragen 33

UPDATE (SQL-Anweisung) 34

UPPER, Funktion 41

URL (Uniform Resource Locator) für den JDBC-Client-Treiber 22

USERNAME, Funktion 41

## V

VAL, Funktion 42

VALUES (SQL-Klausel) 33

Verbindungen, Datenbank 8

## W

WHERE (SQL-Klausel) 29

Wiederholfelder 27, 47

Wildcards mit SELECT-Anweisung 28, 31

Windows

Erstellen eines Datenquellennamens 17

Installation des JDBC-Client-Treibers 13

Installation des ODBC-Client-Treibers 12

JDBC-Client-Treiber 12

ODBC-Client-Treiber-Anforderungen 11

Überprüfen des ODBC-Zugriffs 19

## X

X, Funktion 42

## Y

YEAR, Funktion 42

## Z

Zeichenfolge, Funktionen 41

Zeichenoperatoren in SQL-Ausdrücken 38

Zugriff über ODBC/JDBC, erweitertes

Zugriffsrecht 17

Zugriff überprüfen

JDBC-Client-Treiber 25

ODBC-Client-Treiber (Windows) 19

Zugriffsrechte, erweitert 17