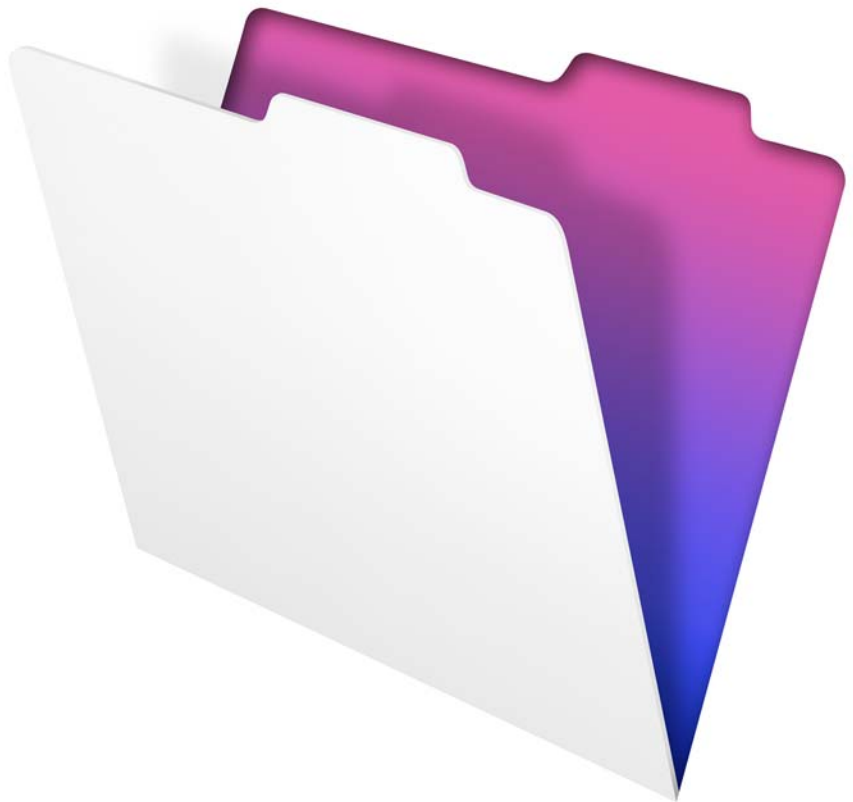


FileMaker® 11

ODBC- und JDBC-Handbuch



© 2004-2010 FileMaker, Inc. Alle Rechte vorbehalten.

FileMaker, Inc.

5201 Patrick Henry Drive

Santa Clara, California 95054, USA

FileMaker ist eine Marke von FileMaker, Inc., eingetragen in den USA und anderen Ländern. Das Dateiodner-Logo ist eine Marke von FileMaker, Inc. Alle anderen Marken sind Eigentum der jeweiligen Besitzer.

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.

Edition: 01

Inhalt

Kapitel 1

Einführung

Über dieses Handbuch	7
ODBC und JDBC	7
Verwenden von FileMaker-Software als ODBC-Client-Anwendung	8
Importieren von ODBC-Daten	8
Hinzufügen von ODBC-Tabellen zum Beziehungsdiagramm	8
Verwenden einer FileMaker-Datenbank als Datenquelle	9
Zugriff auf eine bereitgestellte FileMaker Pro-Datenbank	9
Einschränkungen bei Verwendung von Tools Dritter	10
Netzwerkanforderungen	10
Aktualisieren von Dateien aus früheren Versionen	10
Installation aktueller Treiber	10

Kapitel 2

Zugreifen auf externe SQL-Datenquellen

Importieren von ODBC-Daten	11
Ausführen von SQL zur Interaktion mit Datenquellen über ODBC	12
Arbeiten mit ODBC-Tabellen im Beziehungsdiagramm	13
In FileMaker 11 unterstützte Datenquellen	13
Hinzufügen von ODBC-Tabellen zum Beziehungsdiagramm	13

Kapitel 3

Installation der FileMaker ODBC-Client-Treiber

Hardware- und Software-Anforderungen	15
ODBC-Client-Treiber-Anforderungen (Windows)	15
ODBC-Client-Treiber-Anforderungen (Mac OS)	15
Netzwerkanforderungen	15
ODBC-Client-Treiber-Installation (Windows)	16
Konfigurieren der Client-Treiber (Windows)	16
ODBC-Client-Treiber-Installation (Mac OS)	18
Konfigurieren der Client-Treiber (Mac OS)	18
Die nächsten Schritte	20

Kapitel 4

Verwenden von ODBC zum gemeinsamen Verwenden von FileMaker-Daten

Erläuterung von ODBC	21
Verwenden des ODBC-Client-Treibers	22
Übersicht über den Zugriff auf eine FileMaker-Datenbankdatei	23
Zugreifen auf eine FileMaker-Datenbankdatei von einer Windows-Anwendung aus	24
Festlegen der ODBC-Client-Treibereigenschaften für einen FileMaker-Datenquellennamen (Windows)	24
Überprüfen des Zugriffs über ODBC (Windows)	25
Zugreifen auf eine FileMaker-Datenbankdatei von einer Mac OS-Anwendung aus	26
Festlegen der ODBC-Client-Treibereigenschaften für einen FileMaker-Datenquellennamen (Mac OS)	26
Überprüfen des Zugriffs über ODBC (Mac OS)	27

Kapitel 5

Installieren des FileMaker JDBC-Client-Treibers

Software-Anforderungen	29
Netzwerkanforderungen	29
JDBC-Client-Treiber-Installation	29
Verwenden des JDBC-Client-Treibers	30

Kapitel 6

Verwenden von JDBC zum gemeinsamen Verwenden von FileMaker-Daten

Erläuterung von JDBC	31
Verwenden des JDBC-Client-Treibers	31
Erläuterung des JDBC-Client-Treibers	32
Herstellen einer Verbindung zu Ihrer Datenbank mit einem JDBC-URL	32
Angaben von Treibereigenschaften im URL-Subnamen	34
Lösungen mit mehreren FileMaker-Datenbankdateien	34
Überprüfen des Zugriffs über JDBC	35

Kapitel 7

Unterstützte Standards

Unterstützung von Unicode-Zeichen	37
SQL-Anweisungen	37
SELECT-Anweisung	37
SQL-Klauseln	38
FROM-Klausel	39
WHERE-Klausel	39
GROUP BY-Klausel	40
HAVING-Klausel	40
UNION-Operator	40
ORDER BY-Klausel	41
FOR UPDATE-Klausel	41
DELETE-Anweisung	44
INSERT-Anweisung	44
UPDATE-Anweisung	45
CREATE TABLE-Anweisung	46
ALTER TABLE-Anweisung	46
CREATE INDEX-Anweisung	47
DROP INDEX-Anweisung	47
SQL-Statistikfunktionen:	48
SQL-Ausdrücke	49
Feldnamen	49
Konstanten	49
Exponentialschreibweise	50
Numerische Operatoren	50
Zeichenoperatoren	50
Datumsoperatoren	50
Relationale Operatoren	51
Logische Operatoren	52
Funktionen	53
Funktionen, die Zeichenfolgen zurückgeben	53
Funktionen, die Zahlen zurückgeben	55
Funktionen, die Datumswerte zurückgeben	56
Priorität der Operatoren	56
ODBC-Katalogfunktionen	57
JDBC-Metadaten-Funktionen	57
Reservierte SQL-Schlüsselwörter	57

Kapitel 8

Referenzinformationen

Zuordnen von FileMaker-Feldern zu ODBC-Datentypen	61
Zuordnen von FileMaker-Feldern zu JDBC-Datentypen	61
ODBC- und JDBC-Fehlermeldungen	62
ODBC-Fehlermeldungen	62
JDBC-Fehlermeldungen	62
 <i>Index</i>	 65

Kapitel 1

Einführung

Dieses Handbuch beschreibt, wie Sie FileMaker®-Software als ODBC-Client-Anwendung und als Datenquelle für ODBC- und JDBC-Anwendungen verwenden können.

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

Was wollen Sie erledigen?	Wie wollen Sie vorgehen?	Produkt	Siehe
<ul style="list-style-type: none">Verwenden von FileMaker Pro als ODBC-Client-AnwendungZugriff auf ODBC-Daten, die in einer externen SQL-Datenquelle gespeichert sind.	<ol style="list-style-type: none">Interaktiv über das BeziehungsdiagrammEinmalig statisch per ODBC-Import oder Datei (Windows) bzw. Ablage (Mac OS) > Öffnen. Außerdem der Scriptschritt „Datensätze importieren“ und der Scriptschritt „SQL ausführen“	<ul style="list-style-type: none">FileMaker ProFileMaker Pro AdvancedFileMaker ServerFileMaker Server Advanced	<ul style="list-style-type: none">Dieses Handbuch, Kapitel 2.FileMaker Pro Hilfe
<ul style="list-style-type: none">Verwenden einer FileMaker-Datenbank als DatenquelleGemeinsame Nutzung von FileMaker Pro-Daten mit einer ODBC-Client-Anwendung eines anderen Anbieters.	<ol style="list-style-type: none">SQL-QueriesODBC und JDBC	<ul style="list-style-type: none">FileMaker ProFileMaker Pro AdvancedNur FileMaker Server Advanced	Dieses Handbuch, Kapitel 3 bis 8.

Über dieses Handbuch

- Informationen über die Verwendung von ODBC und JDBC mit früheren Versionen von FileMaker Pro erhalten Sie unter www.filemaker.de/support/product/documentation.html.
- 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.

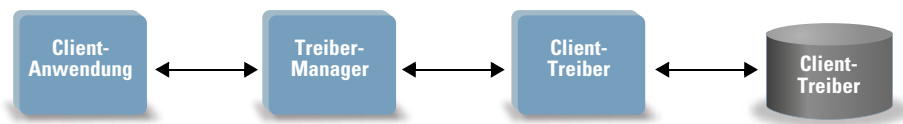
Hinweis Sie können PDF-Dateien der FileMaker-Dokumentation von www.filemaker.de/support/product/documentation.html herunterladen. Aktualisierungen dieses Dokuments erhalten Sie ebenfalls auf der Website.

ODBC und JDBC

ODBC und JDBC sind Schnittstellen zur Programmierung von Anwendungen (APIs). ODBC ist ein API für Anwendungen, die in der Programmiersprache C erstellt wurden, und JDBC ist ein vergleichbares API für Java. 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 (Structured Query Language). Wenn Sie mit SQL arbeiten, können Sie andere Anwendungen (wie Tabellenkalkulationen, Textverarbeitungsprogramme und Bericht-Tools) verwenden, um Daten anzuzeigen, zu analysieren und zu bearbeiten.

Bei Verwendung von ODBC- oder JDBC-APIs kommuniziert eine *Client-Anwendung* mit einem *Treibermanager*, der den *Client-Treiber* identifiziert, um mit einer *Datenquelle* zu kommunizieren.



FileMaker-Software kann entweder als Client-Anwendung oder als Datenquelle fungieren.

Verwenden von FileMaker-Software als ODBC-Client-Anwendung

Als ODBC-Client-Anwendung kann FileMaker-Software auf Daten in externen SQL-Datenquellen zugreifen. FileMaker-Software stellt mittels des Client-Treibers für die ODBC-Datenquelle eine Verbindung zu der externen SQL-Datenquelle her und importiert ODBC-Daten oder arbeitet mit ODBC-Tabellen im Beziehungsdiagramm.

Importieren von ODBC-Daten

Sie können ODBC-Daten auf eine der folgenden Arten importieren:

- über das Menü „Datei“ bzw. „Ablage“, indem Sie eine ODBC-Datenquelle angeben und SQL-Anweisungen in das FileMaker Pro-Dialogfeld für die SQL Query-Erstellung eingeben
- indem Sie ein FileMaker-Script erstellen, das den Scriptbefehl „Datensätze importieren“ oder „SQL Query ausführen“ verwendet.

Für beide Methoden geben Sie die SQL-Anweisungen selbst ein, also müssen Sie die unterstützten SQL-Anweisungen und deren Syntax für Ihre ODBC-Datenquelle kennen. Da Sie die SQL-Anweisungen selbst schreiben, können Sie ODBC-Daten von jeder ODBC-Datenquelle importieren.

Hinzufügen von ODBC-Tabellen zum Beziehungsdiagramm

Wenn Sie dem Beziehungsdiagramm eine ODBC-Tabelle hinzufügen, können Sie eine Verbindung zu externen Datenquellen herstellen und mit den dort enthaltenen Daten arbeiten, ähnlich wie Sie mit Daten in der aktuellen, aktiven FileMaker-Datei arbeiten. Sie können beispielsweise:

- Tabellen für ODBC-Datenquellen im Beziehungsdiagramm erstellen.
- ODBC-Tabellen Ergänzungsfelder hinzufügen, um ungespeicherte Berechnungen durchzuführen oder Daten der ODBC-Tabellen auszuwerten.
- externe Daten interaktiv hinzufügen, ändern und löschen.
- Beziehungen zwischen Feldern in FileMaker-Tabellen und Feldern (so genannten „Spalten“) in ODBC-Tabellen erstellen.

Da FileMaker Pro die SQL-Anweisungen generiert, um mit einer anderen ODBC-Tabelle zu kommunizieren, die dem Beziehungsdiagramm hinzugefügt wurde, sind Sie auf die angegebenen Oracle-, SQL Server- und MySQL-Datenquellen beschränkt, die FileMaker Pro im Beziehungsdiagramm unterstützt.

Hinweis Sie können das Schema externer ODBC-Datenquellen mit FileMaker Pro nicht ändern.

Kapitel 2, „Zugreifen auf externe SQL-Datenquellen“ beschreibt, wie Sie FileMaker Pro-Software als ODBC-Client-Anwendung verwenden.

Verwenden einer FileMaker-Datenbank als Datenquelle

Als Datenquelle werden FileMaker-Daten mit ODBC- und JDBC-fähigen Anwendungen gemeinsam genutzt. Die Anwendung stellt unter Verwendung des FileMaker-Client-Treibers eine Verbindung zur FileMaker-Datenquelle her, erstellt und führt die SQL-Queries mittels ODBC oder JDBC aus und verarbeitet die aus der FileMaker-Datenbanklösung abgerufenen Daten.

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)

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

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.

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 (Open Database Connectivity), JDBC (Java Database Connectivity) und SQL (Structured Query Language) unterstützen.

- Kapitel 3, „Installation der FileMaker ODBC-Client-Treiber“, erläutert wie Sie die für den Zugriff auf eine FileMaker-Datenquelle mittels ODBC benötigten Treiberdateien installieren.
- Kapitel 4, „Verwenden von ODBC zum gemeinsamen Verwenden von FileMaker-Daten“, beschreibt wie Sie den FileMaker ODBC-Client-Treiber verwenden, um von einer anderen Anwendung aus eine Verbindung zu einer FileMaker-Datenquelle herzustellen.
- Kapitel 5, „Installieren des FileMaker JDBC-Client-Treibers“, erläutert wie Sie die für den Zugriff auf eine FileMaker-Datenquelle mittels JDBC benötigten Treiberdateien installieren.
- Kapitel 6, „Verwenden von JDBC zum gemeinsamen Verwenden von FileMaker-Daten“, beschreibt wie Sie den FileMaker JDBC-Client-Treiber mit einer Java-Anwendung oder einem Applet verwenden, das eine Verbindung zu einer FileMaker-Datenquelle herstellt.

- 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 Access: Wenn Sie Microsoft Access verwenden, um Daten in einer FileMaker-Datenquelle anzuzeigen, verwenden Sie nicht die Daten aus einem Statistikfeld. Die Daten des Statistikfelds dürfen nicht in Microsoft Access bearbeitet werden und der in Microsoft Access angezeigte Datenwert könnte auch falsch sein.

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 Sie einen Treiber von früheren Versionen von FileMaker Pro oder FileMaker Server Advanced installiert haben, müssen Sie den Treiber für Version 11 installieren.

Der Treiber für FileMaker Version 11 ist nicht kompatibel zu früheren Versionen von FileMaker Pro oder FileMaker Server Advanced.

Weitere Informationen zur Installation von Treibern finden Sie unter Kapitel 3, „Installation der FileMaker ODBC-Client-Treiber“ und Kapitel 5, „Installieren des FileMaker JDBC-Client-Treibers“.

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

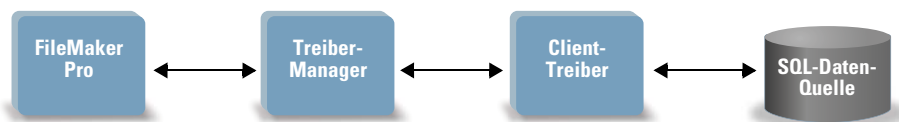
Zugreifen auf externe SQL-Datenquellen

Als ODBC-Client-Anwendung kann FileMaker-Software auf Daten in externen SQL-Datenquellen zugreifen. FileMaker-Software stellt mittels des Client-Treibers für die ODBC-Datenquelle eine Verbindung zu der externen SQL-Datenquelle her und importiert ODBC-Daten oder arbeitet mit ODBC-Tabellen im Beziehungsdiagramm.

Ob Sie ODBC-Daten importieren oder mit ODBC-Tabellen im Beziehungsdiagramm arbeiten, Sie müssen einen Treiber für die verwendete ODBC-Datenquelle konfigurieren. Um z. B. auf Datensätze aus einer Oracle-Datenbank zuzugreifen, konfigurieren Sie einen Oracle-Client-Treiber.

Importieren von ODBC-Daten

Wenn Sie ODBC-Daten importieren, benötigen Sie einen ODBC-Client-Treiber für die auf dem Client-Rechner konfigurierte externe SQL-Datenquelle.



Nach der Konfiguration eines ODBC-Client-Treibers können Sie mit Datensätzen interagieren, Datensätze in eine bestehende FileMaker Pro-Datenbankdatei importieren oder eine neue FileMaker Pro-Datenbankdatei aus einer ODBC-Datenquelle (z. B. Oracle- oder Microsoft Access-Datenbanken) erstellen.

Zuerst greifen Sie auf die Datenquelle zu, aus der Sie importieren wollen. Dann konstruieren Sie eine Query für die Datensätze, die Sie aus der Datenquelle importieren wollen. Wenn Sie Daten in eine bestehende Datei importieren, ordnen Sie schließlich Felder aus Ihrer Datenquelle den Feldern in Ihrer FileMaker Pro-Datenbankdatei zu.

Sie können über das Menü „Datei“ bzw. „Ablage“, über den Scriptschritt „Datensätze importieren“ oder den Scriptschritt „SQL Query ausführen“ auf Ihre ODBC-Datenquelle zugreifen.

Um ODBC-Daten zu importieren, gehen Sie wie folgt vor:

- Installieren und konfigurieren Sie spezifische ODBC-Treiber für die externen Datenquellen, auf die Sie zugreifen wollen.
- Definieren Sie auf dem Computer, der die aktuelle FileMaker Pro-Datei bereitstellt, einen System-DSN (Data Source Name) für jede ODBC-Datenquelle, auf die Sie zugreifen wollen.
- Stellen Sie Optionen für die ODBC-Datenquellen fest, auf die Sie zugreifen wollen (zum Beispiel, ob Benutzer nach einem Benutzernamen und einem Passwort gefragt werden).

- Führen Sie in FileMaker Pro einen der folgenden Schritte aus:
 - Um in eine bestehende FileMaker Pro-Datei zu importieren, wählen Sie Datei (Windows) bzw. > Ablage (Mac OS) > Datensätze importieren > ODBC-Datenquelle.
 - Um eine FileMaker Pro-Datei aus den Datensätzen der Datenquelle anzulegen, wählen Sie Datei (Windows) bzw. Ablage (Mac OS) > Öffnen. Wählen Sie im Dialogfeld „Öffnen“ unter Dateityp (Windows) bzw. Format (Mac OS) die Option ODBC-Datenquelle.

Wählen Sie Ihre Datenquelle, geben Sie Benutzername und Passwort (falls erforderlich) ein und klicken Sie auf OK, um das Dialogfeld „FileMaker Pro SQL Query-Erstellung“ zu öffnen.

Über das Dialogfeld „FileMaker Pro SQL Query-Erstellung“ können Sie eine Query aufbauen. Wählen Sie die Tabelle, aus der Sie importieren möchten, und wählen Sie dann bestimmte Spalten, die Sie in Ihrer SQL-Query verwenden möchten. Verwenden Sie das WHERE-Register, um Suchkriterien aufzubauen, und das ORDER BY-Register, um eine Sortierfolge anzugeben.

Sie können eine SQL-Anweisung auch direkt in das Feld „SQL-Query“ schreiben.

Sie können die Query direkt ausführen oder Sie können die Scriptschritte „Datensätze importieren“ oder „SQL Query ausführen“ verwenden, um eine Query als Teil eines FileMaker-Scripts auszuführen.

Hinweis ODBC-Import, der Scriptschritt „SQL Query ausführen“ und externe SQL-Datenquellen werden nicht in Runtime-Lösungen unterstützt, die mit FileMaker Pro Advanced erstellt wurden.

Weitere Informationen zum Importieren von Daten, Verwenden des Dialogfelds zum Erstellen von SQL-Queries und der Erstellung von FileMaker-Scripts finden Sie in der FileMaker Pro Hilfe.

Ausführen von SQL zur Interaktion mit Datenquellen über ODBC

Zusätzlich zum Importieren von Daten in eine FileMaker Pro-Datenbankdatei über ODBC können Sie mit Datenquellen über SQL-Anweisungen über den Scriptschritt „SQL Query ausführen“ arbeiten. Sie können eine beliebige SQL-Anweisung verwenden, die von der Datenquelle unterstützt wird, z. B. INSERT, UPDATE und DELETE.

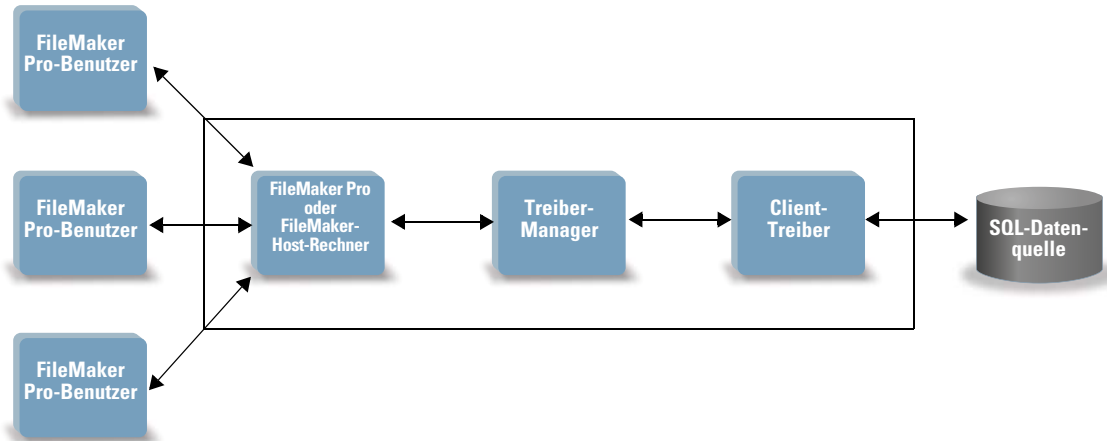
Sie können auch SQL-Anweisungen nutzen, die über den einfachen Datenimport in eine FileMaker Pro-Datenbankdatei hinausgehen. Sie können z. B. SQL-Anweisungen ausführen, die einer Datenbanktabelle in SQL Server Datensätze mit Informationen aus einer FileMaker Pro-Datenbankdatei hinzufügen.

Weitere Informationen über das Erstellen von FileMaker-Scripts, die den Scriptschritt „SQL Query ausführen“ verwenden, finden Sie in der FileMaker Pro Hilfe.

Arbeiten mit ODBC-Tabellen im Beziehungsdiagramm

Wenn Sie dem Beziehungsdiagramm eine ODBC-Tabelle hinzufügen, können Sie eine Verbindung zu externen Datenquellen herstellen und mit den dort enthaltenen Daten arbeiten, ähnlich wie Sie mit Daten in der aktuellen, aktiven FileMaker-Datei arbeiten.

Wenn Sie FileMaker Pro oder FileMaker Server als Host für eine Lösung verwenden, die ODBC-Tabellen im Beziehungsdiagramm enthält, konfigurieren Sie den ODBC-Client-Treiber für die externe SQL-Datenquelle auf dem Hostrechner.



In FileMaker 11 unterstützte Datenquellen

Als ODBC-Client-Anwendung unterstützt FileMaker die folgenden externen SQL-Datenquellen als ODBC-Tabellen im Beziehungsdiagramm:

- Oracle 9i
- Oracle 10g
- Oracle 11g
- SQL Server 2000
- SQL Server 2005
- SQL Server 2008
- MySQL 5.0 Community Edition (kostenlos)
- MySQL 5.1 Community Edition (kostenlos)

Informationen zu unterstützten Client-Treibern finden Sie unter www.filemaker.de/support/technologies.

Hinzufügen von ODBC-Tabellen zum Beziehungsdiagramm

So richten Sie eine FileMaker Pro-Datenbank für den Zugriff auf Daten in unterstützten ODBC-Datenquellen ein:

- Installieren und konfigurieren Sie spezifische ODBC-Treiber für die externen Datenquellen, auf die Sie zugreifen wollen.
- Definieren Sie auf dem Computer, der die aktuelle FileMaker Pro-Datei bereitstellt, einen System-DSN (Data Source Name) für jede ODBC-Datenquelle, auf die Sie zugreifen wollen.

- Stellen Sie Optionen für die ODBC-Datenquellen fest, auf die Sie zugreifen wollen (zum Beispiel, ob Benutzer nach einem Benutzernamen und einem Passwort gefragt werden).
- Fügen Sie dem Beziehungsdiagramm in der aktuellen FileMaker Pro-Datei eine oder mehrere Tabellen aus der ODBC-Datenquelle hinzu.
- Fügen Sie den Layouts in der FileMaker Pro-Datei Felder hinzu, um externe Daten anzuzeigen.
- Fügen Sie externen Tabellen und Layouts optional Ergänzungsfelder hinzu, um Formel- und Statistikergebnisse basierend auf Daten anzuzeigen, die in externen ODBC-Datenquellen gespeichert sind.

Detaillierte Schritte und weitere Informationen zur Konfiguration eines ODBC-Client-Treibers, die Verbindung zu ODBC-Datenquellen, das Bearbeiten von ODBC-Datenquellen und das Einrichten einer ODBC-Datenquelle in einem Beziehungsdiagramm finden Sie in der FileMaker Pro Hilfe.

Kapitel 3

Installation der FileMaker ODBC-Client-Treiber

Diese Anweisungen helfen Ihnen bei der Installation des ODBC-Client-Treibers für den Zugriff auf FileMaker als Datenquelle von Anwendungen Dritter und eigenen Anwendungen über ODBC (Open Database Connectivity). Der ODBC-Client-Treiber steht über ein eigenes Installationsprogramm auf der FileMaker-Installationsdiskette oder im elektronischen Download im Ordner \xDBC zur Verfügung.

Die neuesten Versionen der Client-Treiber stehen auch über diesen URL zur Verfügung:

www.filemaker.de/support/technologies

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.

Hardware- und Software-Anforderungen

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

ODBC-Client-Treiber-Anforderungen (Windows)

- Pentium III 700 MHz oder schneller
- mind. 256 MB RAM, 2 GB RAM empfohlen für Windows XP; mind. 1 GB RAM für Windows Vista und Windows 7
- Microsoft Data Access Components (MDAC), 2.8 SP1
- Windows MDAC 6.0 für Vista

ODBC-Client-Treiber-Anforderungen (Mac OS)

- Apple G4 oder G5; oder Mac OS-Computer mit Intel-Prozessor
- mind. 1 GB RAM; 2 GB RAM empfohlen
- Mac OS X 10.5.7 oder 10.6 (Die Software kann auch mit späteren Versionen verwendet werden, die von FileMaker zertifiziert sind.)

Netzwerkanforderungen

Für den Zugriff auf eine FileMaker-Datenquelle, die auf einem anderen Computer bereitgestellt wird, benötigen Sie einen Netzwerkzugriff über TCP/IP.

ODBC-Client-Treiber-Installation (Windows)

Hinweis Sie benötigen MDAC 2.8 SP1 (verfügbar unter www.microsoft.com), um den ODBC-Client-Treiber auf einem 32-Bit-Windows-Betriebssystem zu installieren.

So installieren Sie den ODBC-Client-Treiber:

1. Sie haben folgende Möglichkeiten:
 - Wenn Sie Ihre Software elektronisch erhalten haben, doppelklicken Sie auf das Installationssymbol (.exe-Datei).
 - Wenn Sie über eine Installationsdiskette verfügen, legen Sie die Diskette in das Laufwerk ein.
2. Doppelklicken Sie im FileMaker Pro- oder FileMaker Server-Fenster auf den Ordner xDBC.
3. Doppelklicken Sie im Ordner xDBC auf den Ordner „ODBC Client Driver Installer“.
4. Doppelklicken Sie im Ordner „ODBC Client Driver Installer“ auf die Datei FMODBC_Installer_Win32.msi. Der Assistent für die FileMaker ODBC-Treiber-Einrichtung wird geöffnet.
5. Installieren Sie den ODBC-Client-Treiber, indem Sie den Bildschirmanweisungen folgen.
6. Wenn die Installation abgeschlossen ist, klicken Sie auf **Schließen**.
Standardmäßig wird der ODBC-Client-Treiber in diesem Ordner installiert: `c:\windows\system32`
Der ODBC-Client-Treiber FileMaker ODBC steht jetzt für die Konfiguration für den Zugriff auf eine FileMaker-Datenquelle bereit.

Konfigurieren der Client-Treiber (Windows)

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 reservieren.

So konfigurieren Sie den ODBC-Client-Treiber:

1. Wählen Sie in der Windows-Systemsteuerung **Verwaltung > Datenquellen (ODBC)**.
 - Unter Windows XP wird „Verwaltung“ in der Kategorie „Leistung und Wartung“ angezeigt.
 - Unter Windows Vista und Windows 7 wird „Verwaltung“ in der Kategorie **System und Sicherheit** angezeigt.Der ODBC-Datenquellenadministrator wird geöffnet.
2. Klicken Sie auf das Register **System-DSN** oder **Benutzer-DSN**.
3. Klicken Sie auf **Hinzufügen**.
Das Dialogfeld „Neue Datenquelle erstellen“ wird geöffnet.

4. Wählen Sie FileMaker ODBC und klicken Sie auf **Fertig stellen**.

Das Dialogfeld „FileMaker DSN-Konfiguration“ wird angezeigt.

5. Klicken Sie auf **Weiter**.**6.** Geben Sie für **Name** einen Namen ein, der für andere Benutzer eine Bedeutung hat, die auf die FileMaker-Datenquelle zugreifen. Geben Sie für **Beschreibung** eine optionale Beschreibung für die FileMaker-Datenquelle ein. Klicken Sie auf **Weiter**.**7.** Für **Host**:

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

Wenn Sie die gemeinsame Nutzung über ODBC/JDBC in der Host-Anwendung aktiviert haben, können Sie **Mit Host verbinden**, um die Namen der verfügbaren Datenbanken abzurufen wählen. Klicken Sie auf **Weiter**.

Klicken Sie andernfalls auf **Fertig stellen**, um Ihre Datenquelleninformationen zu sichern.

8. Wählen Sie für **Datenbank** eine Datenbank aus der Liste der verfügbaren Datenbanken oder geben Sie den Dateinamen der FileMaker-Datenbankdatei ein, die Sie als Datenquelle verwenden.

Hinweis Für Datenbankdateien, die von FileMaker Server Advanced bereitgestellt werden, könnte die Liste der Datenbanken basierend auf der Einstellung **Filtern der Dateianzeige** gefiltert sein. Weitere Informationen finden Sie in der FileMaker Server Hilfe.

Wenn Sie eine spezielle Handhabung internationaler Sprachen erfordern, klicken Sie auf **Sprache erweitert**. Das Dialogfeld „Erweiterte Sprachoptionen“ wird geöffnet.

- Um die Spracheinstellungen automatisch zu erkennen, wählen Sie die Option **Spracheinstellungen für Anwendung automatisch erkennen**.
- Um die Spracheinstellung anzugeben, deaktivieren Sie die Option **Spracheinstellungen für Anwendung automatisch erkennen** und wählen Sie die gewünschte Systemeinstellung.

Wenn Sie eine Protokolldatei für lange laufende Abfragen erstellen möchten, wählen Sie die Option **Lang laufende Abfragen in eine Protokolldatei speichern** und geben Sie den Namen für die Protokolldatei ein.

Klicken Sie auf **Fertig**, um Ihre Datenquelleninformationen zu sichern.

9. Prüfen Sie die Angaben über Ihren FileMaker-DSN.

- Klicken Sie auf **Testen**, um zu überprüfen, ob Sie den ODBC-Client-Treiber korrekt konfiguriert haben, um auf die FileMaker-Datenquelle zugreifen zu können.
Wenn Sie eine Fehlermeldung erhalten, können Sie die Verbindungsinformationen korrigieren. Eventuell müssen Sie auch prüfen, ob die FileMaker-Datenbankdatei bereitgestellt und verfügbar ist, dass das angegebene FileMaker-Konto Berechtigungen mit dem erweiterten Zugriffsrecht „Zugriff über ODBC/JDBC“ verwendet und dass die Host-Anwendung (FileMaker Pro oder FileMaker Server Advanced) für die gemeinsame Nutzung über ODBC/JDBC eingerichtet wurde.
- Klicken Sie auf **Fertig**, um Ihre Datenquelleninformationen zu sichern.

ODBC-Client-Treiber-Installation (Mac OS)

So installieren Sie den ODBC-Client-Treiber:

1. Sie haben folgende Möglichkeiten:
 - Wenn Sie Ihre Software elektronisch erhalten haben, doppelklicken Sie auf das Diskimage-Symbol (.dmg-Datei).
 - Wenn Sie über eine Installationsdiskette verfügen, legen Sie die Diskette in das Laufwerk ein.
2. Doppelklicken Sie im FileMaker Pro- oder FileMaker Server-Fenster auf den Ordner xDBC.
3. Doppelklicken Sie im Ordner xDBC auf den Ordner „ODBC Client Driver Installer“.
4. Doppelklicken Sie im Ordner „ODBC Client Driver Installer“ auf FileMaker ODBC.mpkg.
Das FileMaker ODBC-Treiber-Installationsprogramm wird geöffnet.
5. Installieren Sie den ODBC-Client-Treiber, indem Sie den Bildschirmweisungen folgen.
6. Wenn die Installation abgeschlossen ist, klicken Sie auf **Schließen**.

Der ODBC-Client-Treiber wird in diesem Ordner installiert:
/Library/ODBC

Hinweis Sie können den Installationsordner für den ODBC-Client-Treiber nicht ändern.

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

Konfigurieren der Client-Treiber (Mac OS)

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.

Diese Anweisungen gehen davon aus, dass Sie den ODBC Manager von Actual Technologies installiert haben, der unter www.odbcmanager.net verfügbar ist und ein Freeware-Produkt ist, für das von FileMaker kein Support angeboten wird.

Sie können auch das Apple-eigene ODBC-Administrator Tool für Mac OS X verwenden. Für Mac OS X Version 10.5 wird ODBC-Administrator als Teil des Betriebssystems installiert. Für Mac OS X Version 10.6 steht ODBC-Administrator unter www.apple.com/support zur Verfügung.

Wichtig Wenn Sie einen FileMaker-Client-Treiber konfigurieren, müssen Sie 2399 als Port reservieren.

So konfigurieren Sie den ODBC-Client-Treiber:

1. Starten Sie das ODBC Manager-Dienstprogramm. (Der ODBC Manager ist im Ordner „Dienstprogramme“ im Ordner „Programme“ installiert.)
2. Klicken Sie auf das Register System-DSN oder Benutzer-DSN und klicken Sie auf Hinzufügen.
Das Dialogfeld „Treiber wählen“ wird geöffnet.

3. Wählen Sie FileMaker ODBC und klicken Sie auf OK.

Das Dialogfeld „FileMaker DSN-Konfiguration“ wird angezeigt.

4. Klicken Sie auf Weiter.**5.** Geben Sie für Name einen Namen ein, der für andere Benutzer eine Bedeutung hat, die auf die FileMaker-Datenquelle zugreifen. Geben Sie für Beschreibung eine optionale Beschreibung für die FileMaker-Datenquelle ein. Klicken Sie auf Weiter.**6.** Für Host:

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

Wenn Sie die gemeinsame Nutzung über ODBC/JDBC in der Host-Anwendung aktiviert haben, können Sie Mit Host verbinden, um die Namen der verfügbaren Datenbanken abzurufen wählen. Klicken Sie auf Weiter.

Klicken Sie andernfalls auf Fertig stellen, um Ihre Datenquelleninformationen zu sichern.

7. Wählen Sie für Datenbank eine Datenbank aus der Liste der verfügbaren Datenbanken oder geben Sie den Dateinamen der FileMaker-Datenbankdatei ein, die Sie als Datenquelle verwenden.

Hinweis Für Datenbankdateien, die von FileMaker Server Advanced bereitgestellt werden, könnte die Liste der Datenbanken basierend auf der Einstellung Filtern der Dateianzeige gefiltert sein. Weitere Informationen finden Sie in der FileMaker Server Hilfe.

Wenn Sie eine spezielle Handhabung internationaler Sprachen erfordern, klicken Sie auf Sprache erweitert. Das Dialogfeld „Erweiterte Sprachoptionen“ wird geöffnet.

- Um die Spracheinstellungen automatisch zu erkennen, wählen Sie die Option Spracheinstellungen für Anwendung automatisch erkennen.
- Um die Spracheinstellung anzugeben, deaktivieren Sie die Option Spracheinstellungen für Anwendung automatisch erkennen und wählen Sie die gewünschte Systemeinstellung.

Wenn Sie eine Protokolldatei für lang laufende Abfragen erstellen möchten, wählen Sie die Option Lang laufende Abfragen in eine Protokolldatei speichern und geben Sie den Namen für die Protokolldatei ein.

Klicken Sie auf Fertig, um Ihre Datenquelleninformationen zu sichern.

8. Prüfen Sie die Angaben über Ihren FileMaker-DSN.

- Klicken Sie auf Testen, um zu überprüfen, ob Sie den ODBC-Client-Treiber korrekt konfiguriert haben, um auf die FileMaker-Datenquelle zugreifen zu können.
Wenn Sie eine Fehlermeldung erhalten, können Sie die Verbindungsinformationen korrigieren. Eventuell müssen Sie auch prüfen, ob die FileMaker-Datenbankdatei bereitgestellt und verfügbar ist, dass das angegebene FileMaker-Konto Berechtigungen mit dem erweiterten Zugriffsrecht „Zugriff über ODBC/JDBC“ verwendet und dass die Host-Anwendung (FileMaker Pro oder FileMaker Server Advanced) für die gemeinsame Nutzung über ODBC/JDBC eingerichtet wurde.
- Klicken Sie auf Fertig, um Ihre Datenquelleninformationen zu sichern.

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.

Weitere Informationen zur Verwendung von FileMaker als ODBC-Datenquelle finden Sie unter Kapitel 4, „Verwenden von ODBC zum gemeinsamen Verwenden von FileMaker-Daten“.

Kapitel 4

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.

Der FileMaker ODBC-Client-Treiber ist FileMaker ODBC.

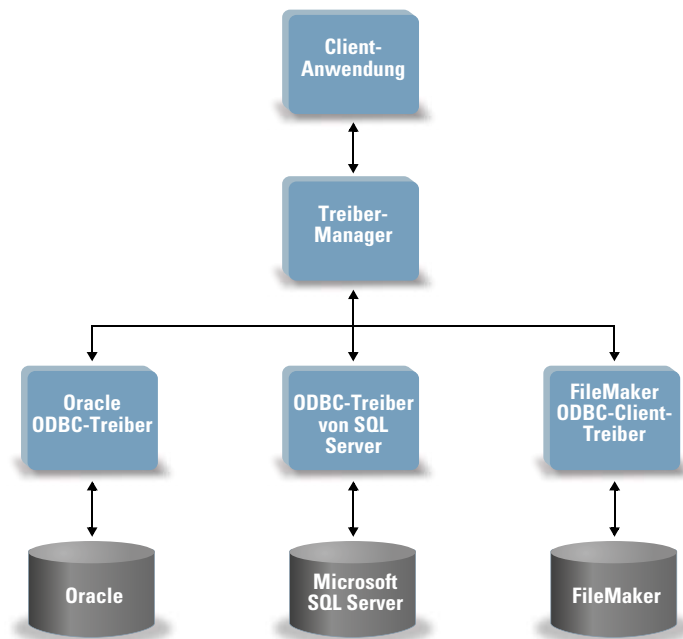
Hinweis Sie können auch FileMaker Pro als ODBC-Client-Anwendung verwenden, die mithilfe von SQL über ODBC mit den Datensätzen einer anderen Datenquelle interagiert. Weitere Informationen über den Zugriff auf eine externe SQL-Datenquelle über ODBC finden Sie in Kapitel 2, „Zugreifen auf externe SQL-Datenquellen“.

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 SQL-Anweisungen (Structured Query Language). 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 7, „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 Frühere Versionen des FileMaker ODBC-Client-Treibers sind nicht kompatibel mit FileMaker Version 11. Um eine Verbindung zu einer FileMaker-Datenbankdatei der Version 11 herzustellen, müssen Sie den neuen ODBC-Client-Treiber installieren und konfigurieren.

Hinweis Um zu bestätigen, dass der FileMaker xDBC Listener aktuell läuft, können Sie unter Mac OS die Aktivitäts-Anzeige oder unter Windows den Task-Manager verwenden, um den Status des FileMaker xDBC Listener-Prozesses zu prüfen. Wenn der Prozess gestartet wurde, heißt er fmxdbc_listener und Ereignisse werden mit diesem Namen protokolliert. Der FileMaker xDBC Listener-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 (Mac 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.

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 File-Maker-Host-Anwendung und die FileMaker-Datenbankdatei, auf die Sie als Datenquelle zugreifen.

Um einen neuen ODBC-Client-Treiber zu konfigurieren, siehe „Konfigurieren der Client-Treiber (Windows)“ auf Seite 16.

So ändern Sie einen vorhandenen ODBC-Client-Treiber:

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

- Unter Windows XP wird „Verwaltung“ in der Kategorie „Leistung und Wartung“ angezeigt.
- Unter Windows Vista und Windows 7 wird „Verwaltung“ in der Kategorie **System und Sicherheit** angezeigt.

Der ODBC-Datenquellenadministrator wird geöffnet.

2. Wählen Sie das Register **System-DSN** bzw. **Benutzer-DSN**- (Wählen Sie das Register, das Sie bei der vorherigen Konfiguration verwendet hatten.)

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

Der ursprünglich eingegebene Datenquellename wird unter **Name** angezeigt und **FileMaker ODBC** wird als **Treiber** angezeigt.

4. Klicken Sie auf **Konfigurieren**.

Das Dialogfeld „FileMaker DSN-Konfiguration“ wird angezeigt.

5. Geben Sie für **Name** einen Namen ein, der für andere Benutzer eine Bedeutung hat, die auf die FileMaker-Datenquelle zugreifen. Geben Sie für **Beschreibung** eine optionale Beschreibung für die FileMaker-Datenquelle ein. Klicken Sie auf **Weiter**.

6. Geben Sie für **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.

Wenn Sie die gemeinsame Nutzung über ODBC/JDBC in der Host-Anwendung aktiviert haben, wählen Sie **Mit Host verbinden**, um die Namen der verfügbaren Datenbanken abzurufen.

- Wählen Sie für **Datenbank** eine Datenbank aus der Liste der verfügbaren Datenbanken oder geben Sie den Dateinamen der FileMaker-Datenbankdatei ein, die Sie als Datenquelle verwenden.

Hinweis Für Datenbankdateien, die von FileMaker Server Advanced bereitgestellt werden, könnte die Liste der Datenbanken basierend auf der Einstellung **Filtern der Dateianzeige** gefiltert sein. Weitere Informationen finden Sie in der FileMaker Server Hilfe.

Wenn Sie eine spezielle Handhabung internationaler Sprachen erfordern, klicken Sie auf **Sprache erweitert**. Das Dialogfeld „Erweiterte Sprachoptionen“ wird geöffnet.

- Um die Spracheinstellungen automatisch zu erkennen, wählen Sie die Option **Spracheinstellungen für Anwendung automatisch erkennen**.
- Um die Spracheinstellung anzugeben, deaktivieren Sie die Option **Spracheinstellungen für Anwendung automatisch erkennen** und wählen Sie die gewünschte Systemeinstellung.

Wenn Sie eine Protokolldatei für lang laufende Abfragen erstellen möchten, wählen Sie die Option **Lang laufende Abfragen in eine Protokolldatei speichern** und geben Sie den Namen für die Protokolldatei ein.

- Klicken Sie auf **Fertig**, um Ihre Datenquelleninformationen zu sichern.
- Klicken Sie auf **Fertig**, um das Dialogfeld „FileMaker DSN-Konfiguration“ 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:

- Wählen Sie in der Windows-Systemsteuerung **Verwaltung > Datenquellen (ODBC)**.
 - Unter Windows XP wird „Verwaltung“ in der Kategorie „Leistung und Wartung“ angezeigt.
 - Unter Windows Vista und Windows 7 wird „Verwaltung“ in der Kategorie **System und Sicherheit** angezeigt.

Der ODBC-Datenquellenadministrator wird geöffnet.

- Wählen Sie das Register **System-DSN** bzw. **Benutzer-DSN**. (Wählen Sie das Register, das Sie bei der vorherigen Konfiguration verwendet hatten.)
- Wählen Sie die FileMaker-Datenquelle, die Sie zuvor konfiguriert haben.

Der ursprünglich eingegebene Datenquellenname wird unter **Name** angezeigt und **FileMaker ODBC** wird als **Treiber** angezeigt.
- Klicken Sie auf **Konfigurieren**.

Das Dialogfeld „FileMaker DSN-Konfiguration“ wird angezeigt.
- Klicken Sie auf **Weiter**, bis Sie zur Seite **Ergebnis** gelangen.
- Klicken Sie auf **Testen**.

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

Falls die Verbindung OK ist, wird die Meldung **Test erfolgreich abgeschlossen** 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

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.

Um einen neuen ODBC-Client-Treiber zu konfigurieren, siehe „Konfigurieren der Client-Treiber (Mac OS)“ auf Seite 18.

So ändern Sie einen vorhandenen ODBC-Client-Treiber:

1. Starten Sie das ODBC Manager-Dienstprogramm. (Der ODBC Manager ist im Ordner „Dienstprogramme“ im Ordner „Programme“ installiert.)
2. Klicken Sie auf das Register System-DSN oder Benutzer-DSN.
3. Wählen Sie die FileMaker-Datenquelle, die Sie zuvor konfiguriert haben.
Der ursprünglich eingegebene Datenquellename wird unter Name angezeigt und FileMaker ODBC wird als Treiber angezeigt.
4. Klicken Sie auf Konfigurieren.
Das Dialogfeld „FileMaker DSN-Konfiguration“ wird angezeigt.
5. Klicken Sie auf Weiter.
6. Geben Sie für Name einen Namen ein, der für andere Benutzer eine Bedeutung hat, die auf die FileMaker-Datenquelle zugreifen.
Eine zusätzliche Beschreibung ist optional.
7. Geben Sie für 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.

Wenn Sie die gemeinsame Nutzung über ODBC/JDBC in der Host-Anwendung aktiviert haben, wählen Sie Mit Host verbinden, um die Namen der verfügbaren Datenbanken abzurufen.

8. Wählen Sie für **Datenbank** eine Datenbank aus der Liste der verfügbaren Datenbanken oder geben Sie den Dateinamen der FileMaker-Datenbankdatei ein, die Sie als Datenquelle verwenden.

Hinweis Für Datenbankdateien, die von FileMaker Server Advanced bereitgestellt werden, könnte die Liste der Datenbanken basierend auf der Einstellung **Filtern der Dateianzeige** gefiltert sein. Weitere Informationen finden Sie in der FileMaker Server Hilfe.

Wenn Sie eine spezielle Handhabung internationaler Sprachen erfordern, klicken Sie auf **Sprache erweitert**. Das Dialogfeld „Erweiterte Sprachoptionen“ wird geöffnet.

- Um die Spracheinstellungen automatisch zu erkennen, wählen Sie die Option **Spracheinstellungen für Anwendung automatisch erkennen**.
- Um die Spracheinstellung anzugeben, deaktivieren Sie die Option **Spracheinstellungen für Anwendung automatisch erkennen** und wählen Sie die gewünschte Systemeinstellung.

Wenn Sie eine Protokolldatei für lang laufende Abfragen erstellen möchten, wählen Sie die Option **Lang laufende Abfragen in eine Protokolldatei speichern** und geben Sie den Namen für die Protokolldatei ein.

9. Klicken Sie auf **Fertig**, um Ihre Datenquelleninformationen zu sichern.
10. Klicken Sie auf **Fertig**, um das Dialogfeld „FileMaker DSN-Konfiguration“ zu schließen.

Überprüfen des Zugriffs über ODBC (Mac OS)

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

1. Starten Sie das ODBC Manager-Dienstprogramm. (Der ODBC Manager befindet sich im Ordner „Dienstprogramme“ im Ordner „Programme“.)
2. Wählen Sie das Register **System-DSN** bzw. **Benutzer-DSN**. (Wählen Sie das Register, das Sie bei der vorherigen Konfiguration verwendet hatten.)
3. Wählen Sie die FileMaker-Datenquelle, die Sie zuvor konfiguriert haben.
Der ursprünglich eingegebene Datenquellename wird unter **Name** angezeigt und FileMaker ODBC wird als **Treiber** angezeigt.
4. Klicken Sie auf **Konfigurieren**.
Das Dialogfeld „FileMaker DSN-Konfiguration“ wird angezeigt.
5. Klicken Sie auf **Weiter**, bis Sie erneut zur Seite **Ergebnis** gelangen.
6. Klicken Sie auf **Testen**.
Sie werden aufgefordert, Ihren FileMaker-Kontonamen (unter **Datenbankbenutzername**) und Ihr Passwort (unter **Datenbankpasswort**) einzugeben.

Falls die Verbindung OK ist, wird die Meldung **Test erfolgreich abgeschlossen** 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.

Kapitel 5

Installieren des FileMaker JDBC-Client-Treibers

Diese Anweisungen helfen Ihnen bei der Installation des Client-Treibers für den Zugriff auf FileMaker als Datenquelle von Anwendungen Dritter und eigenen Anwendungen über JDBC (Java Database Connectivity). Der Client-Treiber ist auf Ihrer FileMaker-DVD oder bei elektronischem Download im Ordner xDBC verfügbar. Die neuesten Versionen der Client-Treiber stehen auch über diesen URL zur Verfügung:

www.filemaker.de/support/technologies

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.

Der JDBC-Client-Treiber ist der Treiberbestandteil der FileMaker-Software, die es Anwendungen Dritter oder von Ihnen entwickelten Anwendungen ermöglicht, auf FileMaker-Dateien als JDBC-Datenquelle zuzugreifen.

Software-Anforderungen

Um die JDBC-Client-Treiber zu installieren und zu verwenden, benötigen Sie JDK 1.4 oder höher.

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.

JDBC-Client-Treiber-Installation

Sie benötigen Schreibzugriff für den Ordner, in dem Sie den JDBC-Client-Treiber installieren.

So installieren Sie den JDBC-Client-Treiber:

1. Sie haben folgende Möglichkeiten:

- Windows: Wenn Sie Ihre Software elektronisch erhalten haben, doppelklicken Sie auf das Installationssymbol (.exe-Datei).
- Mac OS: Wenn Sie Ihre Software elektronisch erhalten haben, doppelklicken Sie auf das Diskimage-Symbol (.dmg-Datei).
- Wenn Sie über eine Installationsdiskette verfügen, legen Sie die Diskette in das Laufwerk ein.

2. Doppelklicken Sie im FileMaker Pro- oder FileMaker Server-Fenster auf den Ordner xDBC.

3. Doppelklicken Sie im Ordner xDBC auf den Ordner „JDBC Client Driver Installer“.

4. Kopieren Sie die Datei `fmjdbc.jar` in den entsprechenden Ordner für Ihr Betriebssystem:

- Windows: Kopieren Sie die Datei `fmjdbc.jar` in den Ordner, der Ihre ausführbare Java-Datei (`java.exe`) enthält, oder einen anderen Ordner, der im ClassPath Ihrer Java-Anwendung enthalten ist.
- Mac OS: Kopieren Sie die Datei `fmjdbc.jar` in den Ordner `/Library/Java/Extensions` oder einen anderen Ordner, der im ClassPath Ihrer Java-Anwendung enthalten ist.

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

Verwenden des JDBC-Client-Treibers

Ihre Java-Anwendung bzw. das Java-Applet muss den JDBC-Client-Treiber beim JDBC-Treiber-Manager registrieren und Sie müssen in der Anwendung bzw. im Applet den korrekten JDBC-URL angeben.

Wichtig Sie müssen Port 2399 für den FileMaker JDBC-Client-Treiber reservieren. Die Port-Nummer lautet immer 2399 (für JDBC-Sharing kann die Port-Nummer nicht geändert werden).

Weitere Informationen zur Verwendung des JDBC-Client-Treibers finden Sie unter Kapitel 6, „Verwenden von JDBC zum gemeinsamen Verwenden von FileMaker-Daten“.

Kapitel 6

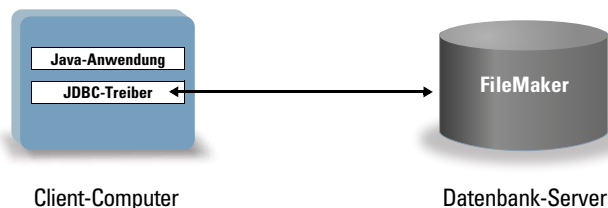
Verwenden von JDBC zum gemeinsamen Verwenden von FileMaker-Daten

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
- Abruf und Aktualisierung des von einem REF-Objekt referenzierten Objekts
- Aktualisierung von Spalten mit 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.

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).

Die Treiberklasse und Haupteinstiegspunkt des Treibers:

```
com.filemaker.jdbc.Driver
```

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 FileMaker xDBC Listener aktuell läuft, können Sie unter Mac OS die Aktivitäts-Anzeige oder unter Windows den Task-Manager verwenden, um den Status des FileMaker xDBC Listener-Prozesses zu prüfen. Wenn der Prozess gestartet wurde, heißt er `fmjdbc_listener` und Ereignisse werden mit diesem Namen protokolliert. Der FileMaker xDBC Listener-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 `filemaker`. 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. Stellt eine Verbindung mit der FileMaker-Datenquelle her. Der JDBC-URL ist `jdbc:filemaker://192.168.1.1/database`

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.filemaker.jdbc.Driver").newInstance();
        } catch (Exception e) {
            System.out.println(e);
        }
        // eine Verbindung mit FileMaker herstellen
        Connection con;
        try {
            con = DriverManager.getConnection("jdbc:filemaker://192.168.1.1/mydatabase","benutzername",
            "passwort");
        } catch (Exception e) {
            System.out.println(e);
        }
        // Verbindungswarnungen holen
        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) {
            System.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

JDBC-URL-Verbindung mit in der URL angegebenem Datenbanknamen

Format:

```
jdbc:filemaker://<Filemaker-Host-IP-Adresse>/<Datenbankname>
```

Beispiel:

```
jdbc:filemaker://192.168.1.1/publications
```

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

Format:

```
jdbc:filemaker://<Filemaker-Host-IP-Adresse>/  
<Datenbankname>?user=<Datenbankbenutzername>&password=<Datenbankpasswort>
```

Beispiel:

```
jdbc:filemaker://192.168.1.1/customers?user=Collections&password=admin
```

Hinweis Aufgrund der Verwendung des &-Zeichens in dieser Syntax können Sie kein &-Zeichen in Benutzernamen oder Passwort verwenden.

Beispiel für einen ungültigen Benutzernamen:

```
jdbc:filemaker://localhost/sales_db?user=ad&min&password=admin
```

Beispiel für ein ungültiges Passwort:

```
jdbc:filemaker://localhost/sales_db?user=admin1&password=ad&min
```

Lösungen mit mehreren FileMaker-Datenbankdateien

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.

Überprüfen des Zugriffs über JDBC

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.

Kapitel 7

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 finden Sie unter www.filemaker.de/support/technologies. 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`.

SQL-Anweisungen

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

SELECT (siehe unten)	DELETE (Seite 44)	INSERT (Seite 44)	UPDATE (Seite 45)
CREATE TABLE (Seite 46)	ALTER TABLE (Seite 46)	CREATE INDEX (Seite 47)	DROP INDEX (Seite 47)

Die Client-Treiber unterstützen darüber hinaus die Zuordnung von FileMaker-Datentypen zu ODBC SQL- und JDBC SQL-Datentypen. Siehe „Zuordnen von FileMaker-Feldern zu ODBC-Datentypen“ auf Seite 61 und „Zuordnen von FileMaker-Feldern zu JDBC-Datentypen“ auf Seite 61 für Datentypkonvertierungen. Zusätzliche Informationen über das Erstellen von SQL-Abfragen finden Sie in Büchern von Fremdanbietern.

Hinweis Die ODBC- und JDBC-Client-Treiber unterstützen keine FileMaker-Ausschnitte.

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.

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	Ziel
FROM (Seite 39)	anzuzeigen, welche Tabellen in der SELECT-Anweisung verwendet werden
WHERE (Seite 39)	die Bedingungen anzugeben, die Datensätze erfüllen müssen, um abgefragt zu werden (wie FileMaker Pro-Suchabfragen)
GROUP BY (Seite 40)	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 40)	die Bedingungen für Gruppen von Datensätzen anzugeben (z. B. um nur die Abteilungen anzuzeigen, die Gehälter von insgesamt mehr als 200.000 Euro haben).
UNION (Seite 39)	die Ergebnisse zweier oder mehrerer SELECT-Anweisungen in einem einzigen Ergebnis zu kombinieren.
ORDER BY (Seite 41)	anzuzeigen, wie die Datensätze sortiert sind.
FOR UPDATE (Seite 41)	„Positioned Updated“ und „Positioned Deletes“ über SQL-Cursor durchzuführen

Hinweis Wenn Sie versuchen, Daten von einer Tabelle ohne Spalten abzurufen, gibt die SELECT-Anweisung nichts zurück.

FROM-Klausel

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

```
FROM tabellenname [tabellenalias], tabellenname [tabellenalias]
```

tabellenname ist der Name einer Tabelle in der aktuellen Datenbank.

tabellenalias kann verwendet werden, um der Tabelle einen beschreibenderen Namen zu geben, einen langen Tabellennamen anzukürzen oder die gleiche Tabelle mehr als einmal in die Query aufzunehmen (z. B. bei Self-Joins).

Der Tabellename 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 Vertriebsdaten
ON Verkaeufer.Verkaeuferrnr = Vertriebsdaten.Verkaeuferrnr
```

Wenn Sie zwei Tabellen verbinden, aber Zeilen der ersten Tabelle (die „linke“ Tabelle) nicht verwerfen möchten, können Sie LEFT JOIN verwenden.

```
SELECT *
FROM Verkaeufer LEFT JOIN Umsatzdaten
ON Verkaeufer.Verkaeufer-ID = Vertriebsdaten.AbcteilungID
```

Jede Zeile aus der Tabelle „Verkaeufer“ erscheint in der verbundenen Tabelle.

Wenn Sie zwei Tabellen verbinden, aber Zeilen der zweiten Tabelle (die „rechte“ Tabelle) nicht verwerfen möchten, können Sie RIGHT JOIN verwenden.

```
SELECT *
FROM Verkaeufer RIGHT JOIN Umsatzdaten
ON Verkaeufer.Verkaeufer-ID = Vertriebsdaten.AbcteilungID
```

Jede Zeile aus der Tabelle „Vertriebsdaten“ erscheint in der verbundenen Tabelle.

Hinweise

- LEFT JOIN wird unterstützt, aber die Grammatik LEFT OUTER JOIN wird aktuell nicht unterstützt.
- RIGHT JOIN wird unterstützt, aber die Grammatik RIGHT OUTER JOIN wird aktuell nicht unterstützt.
- FULL 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
```

Die WHERE-Klausel kann auch Ausdrücke wie diese verwenden:

```
WHERE expr1 IS NULL
WHERE NOT expr2
```

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 Spalten
```

Spalten muss mit dem in der SELECT-Anweisung verwendeten Spaltenausdruck übereinstimmen. Ein Spaltenausdruck kann ein oder mehrere Feldnamen der Datenbanktabelle, 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). 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, einst_datum 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 sperrt Datensätze für „Positioned Updates“ oder „Positioned Deletes“ über SQL-Cursor. 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 und wird ignoriert.

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 freigegeben, wenn Sie den nächsten Datensatz abrufen.

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

Weitere Beispiele:

Verwendung	Beispiel-SQL
Textkonstante	<code>SELECT 'KatzeHund' FROM Verkaeufer</code>
Zahlenkonstante	<code>SELECT 999 FROM Verkaeufer</code>
Datumskonstante	<code>SELECT DATE '05.06.11' FROM Verkaeufer</code>
Zeitkonstante	<code>SELECT TIME '02:49:03' FROM Verkaeufer</code>
Zeitstempelkonstante	<code>SELECT TIMESTAMP '05.06.11 02:49:03' FROM Verkaeufer</code>
Textspalte	<code>SELECT Firmenname FROM Vertriebsdaten</code> <code>SELECT DISTINCT Firmenname FROM Vertriebsdaten</code>
Zahlenspalte	<code>SELECT Betrag FROM Vertriebsdaten</code> <code>SELECT DISTINCT Betrag FROM Vertriebsdaten</code>
Datumsspalte	<code>SELECT Verkaufsdatum FROM Vertriebsdaten</code> <code>SELECT DISTINCT Verkaufsdatum FROM Vertriebsdaten</code>
Zeitspalte	<code>SELECT Verkaufszeit FROM Vertriebsdaten</code> <code>SELECT DISTINCT Verkaufszeit FROM Vertriebsdaten</code>
Zeitstempelspalte	<code>SELECT Verkaufszeitstempel FROM Vertriebsdaten</code> <code>SELECT DISTINCT Verkaufszeitstempel FROM Vertriebsdaten</code>
BLOB ^a -Spalte	<code>SELECT Firmenbroschueren FROM Vertriebsdaten</code> <code>SELECT GETAS(Firmenlogo, 'JPEG') FROM Vertriebsdaten</code>
Jokerzeichen *	<code>SELECT * FROM Verkaeufer</code> <code>SELECT DISTINCT * FROM Verkaeufer</code>

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 Verkaeufer` ist das Ergebnis alle Spalten in der Tabelle `Verkaeufer`. Für das Beispiel `SELECT DISTINCT * FROM Verkaeufer` ist das Ergebnis alle eindeutigen Zeilen in der Tabelle `Verkaeufer` (keine doppelten Werte).

- FileMaker speichert keine Daten für leere Zeichenfolgen, so dass die folgenden Queries immer keine Datensätze zurückgeben:

```
SELECT * FROM test WHERE c = ''
SELECT * FROM test WHERE c <> ''
```

- Wenn Sie `SELECT` mit Binärdaten verwenden, müssen Sie die Funktion `GetAs()` verwenden, um den zurückzugebenden Stream anzugeben. Weitere Informationen finden Sie im folgenden Abschnitt, „Abrufen des Inhalts eines Medienfelds: `CAST()`-Funktion und `GetAs()`-Funktion“.

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(Firmenbroschuere 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 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
'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, ...)
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 Werteausdrücke (*ausdr*) Werte für alle in der Tabelle definierten Spalten angeben und in der gleichen Reihenfolge sein wie die für die Tabelle definierten Spalten. *spaltenname* kann auch eine Feldwiederholung, zum Beispiel `lastDates[4]` angeben.

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 einfachen 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`).

Unterabfragen müssen in Klammern angegeben werden.

Das folgende Beispiel fügt eine Liste von Ausdrücken ein:

```
INSERT INTO ang (nachname, vorname, angnr, gehalt, einst_datum)
VALUES ('Schmidt', 'Johann', 'E22345', 27500, {d '05.06.2008'})
```

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 Medienfeldern können Sie nur Text EINFÜGEN, wenn Sie keine parametrisierte Anweisung vorbereiten und die Daten aus Ihrer Anwendung streamen. Um Binärdaten zu verwenden, müssen Sie den Typ in einer PutAs()-Funktion angeben: `PutAs(col, 'typ')`, wobei der Typwert ein Typ wobei der Typwert ein Typ ist, der unter „Abrufen des Inhalts eines Medienfelds: CAST()-Funktion und GetAs()-Funktion“ auf Seite 42 beschrieben wird.

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 einfachen 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`).

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 Gehaltswert des Unternehmens geändert.

Hinweis In Medienfeldern können Sie nur Text AKTUALISIEREN, wenn Sie keine parametrisierte Anweisung vorbereiten und die Daten aus Ihrer Anwendung streamen. Um Binärdaten zu verwenden, müssen Sie den Typ in einer PutAs()-Funktion angeben: `PutAs(col, 'typ')`, wobei der Typwert ein Typ wobei der Typwert ein Typ ist, der unter „Abrufen des Inhalts eines Medienfelds: CAST()-Funktion und GetAs()-Funktion“ auf Seite 42 beschrieben wird.

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 [DEFAULT ausdr] [UNIQUE] [NOT NULL]
```

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

- tabellenname und tabellenelementliste sind auf 100 Zeichen beschränkt.
- Über das Schlüsselwort DEFAULT können Sie einen Standardwert für eine Spalte festlegen. Sie können ein Literal wie CURRENT_USER, CURRENT_DATE, CURRENT_TIME oder CURRENT_TIMESTAMP verwenden.
- Die Definition einer Spalte als UNIQUE wählt automatisch die Überprüfungsoption Eindeutig für das entsprechende Feld in der FileMaker-Datenbankdatei aus.
- 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
ALTER TABLE tabellenname ALTER [COLUMN] spaltendefinition SET DEFAULT ausdr
ALTER TABLE tabellenname ALTER [COLUMN] spaltendefinition DROP DEFAULT
```

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

Beispiele

Ziel	Beispiel-SQL
Spalten hinzuzufügen	<code>ALTER TABLE Verkaeufner ADD C1 VARCHAR</code>
Spalten zu entfernen	<code>ALTER TABLE Verkaeufner DROP C1</code>
Legt den Standardwert für eine Spalte fest.	<code>ALTER TABLE Verkaeufner ALTER Firma SET DEFAULT 'FileMaker'</code>
Entfernt den Standardwert für eine Spalte.	<code>ALTER TABLE Verkaeufner ALTER Firma DROP DEFAULT</code>

Hinweis `SET DEFAULT` und `DROP DEFAULT` wirken sich nicht auf vorhandene Zeilen in der Tabelle aus, aber ändern den Standardwert für Zeilen, die später der Tabelle hinzugefügt werden.

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 ON tabellenname.spaltenname
CREATE INDEX 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.

FileMaker erstellt Indizes bei Bedarf automatisch. Die Verwendung von `CREATE INDEX` bewirkt, dass der Index direkt als nur bei Bedarf erstellt wird.

Beispiel:

```
CREATE INDEX ON Verkaeufner.VerkaeufnerID
```

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
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 Verkaeuf.er.Verkaeuf.ernr
```

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.

Beispiele

Statistikfunktion	Retouren
SUM	Die Summe der Werte in einem Zahlenfeldausdruck. Beispiel: SUM(GEHALT) gibt die Summe aller Gehaltsfeldwerte zurück.
AVG	Der Mittelwert der Werte in einem Zahlenfeldausdruck. Beispiel: AVG(GEHALT) gibt den Mittelwert aller Gehaltsfeldwerte zurück.
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.

```
SELECT SUM (Vertriebsdaten.Betrag) AS agg FROM Vertriebsdaten
SELECT AVG (Vertriebsdaten.Betrag) AS agg FROM Vertriebsdaten
SELECT COUNT (Vertriebsdaten.Betrag) AS agg FROM Vertriebsdaten
SELECT MAX (Vertriebsdaten.Betrag) AS agg FROM Vertriebsdaten
WHERE Vertriebsdaten.Betrag < 3000
SELECT MIN (Vertriebsdaten.Betrag) AS agg FROM Vertriebsdaten
WHERE Vertriebsdaten.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	Zeichenoperatoren	Logische Operatoren
Exponentialschreibweise	Datumsoperatoren	Funktionen

Feldnamen

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

Konstanten

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.

Sie müssen Zeichenkonstanten in einfachen 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`).

FileMaker akzeptiert die Konstanten für ODBC/JDBC-Formatdatum, -Zeit und -Zeitstempel in geschweiften Klammern ({}), zum Beispiel:

- `{D '05.06.2010'}`
- `{T '14:35:10'}`
- `{TS '05.06.2010 14:35:10'}`

FileMaker akzeptiert auch die ISO-Datums- und Zeitformate der SQL-92-Syntax ohne geschweifte Klammern:

- `DATE 'JJJJ-MM-TT'`
- `TIME 'HH:MM:SS'`
- `TIMESTAMP 'JJJJ-MM-TT HH:MM:SS'`

Konstante	Akzeptable Syntax (Beispiele)
Text	'Paris'
Zahl	1.05
Datum	DATE '05.06.2010' { D '05.06.2010' } {06/05/2010} {06/05/10}

Hinweis: Die 2-stellige Jahressyntax wird nicht für das ODBC/JDBC-Format oder das SQL-92-Format unterstützt.

Konstante	Akzeptable Syntax (Beispiele)
Zeit	TIME '14:35:10' { T '14:35:10' } { 14:35:10 }
Zeitstempel	TIMESTAMP '05.06.2010 14:35:10' { TS '05.06.2010 14:35:10' } { 06/05/2010 14:35:10 } { 06/05/10 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. Hinweis: Die 2-stellige Jahressyntax wird nicht für das ODBC/JDBC-Format oder das SQL-92-Format unterstützt.

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.

Exponentialschreibweise

Zahlen können auch in wissenschaftlicher Schreibweise angegeben werden.

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 `einst_datum` {D '20.01.2008'}.

Operator	Wirkung auf Datum	Beispiel:	Ergebnis
+	Einem Datum eine Anzahl von Tagen hinzufügen	<code>einst_datum + 5</code>	{D '02.04.2008'}
-	Die Anzahl von Tagen zwischen zwei Datumswerten ermitteln oder eine Anzahl von Tagen von einem Datum abziehen	<code>einst_datum - {D '01.01.2008'}</code> <code>einst_datum - 10</code>	29 {D '20.01.2008'}

Weitere Beispiele:

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

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

```

SELECT Vertriebsdaten.Verkaufszeit FROM Vertriebsdaten
  WHERE Vertriebsdaten.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 (Vertriebsdaten.Rechnungsnr) AS agg
  FROM Vertriebsdaten WHERE Vertriebsdaten.Rechnungsnr IN (50,250,100)
SELECT COUNT (Vertriebsdaten.Rechnungsnr) AS agg
  FROM Vertriebsdaten WHERE Vertriebsdaten.Rechnungsnr NOT IN (50,250,100)
SELECT COUNT (Vertriebsdaten.Rechnungsnr) AS agg FROM Vertriebsdaten
  WHERE Vertriebsdaten.Rechnungsnr NOT IN (SELECT
Vertriebsdaten.Rechnungsnr
  FROM Vertriebsdaten WHERE Vertriebsdaten.Verkaeufernr = 'SP-4')
SELECT *
  FROM Vertriebsdaten WHERE EXISTS (SELECT Vertriebsdaten.Betrag
  FROM Vertriebsdaten WHERE Vertriebsdaten.Verkaeufernr IS NOT NULL)
SELECT *
  FROM Vertriebsdaten WHERE Vertriebsdaten.Betrag = ANY (SELECT
Vertriebsdaten.Betrag
  FROM Vertriebsdaten WHERE Vertriebsdaten.Verkaeufernr = 'SP-1')
SELECT *
  FROM Vertriebsdaten WHERE Vertriebsdaten.Betrag = ALL (SELECT
Vertriebsdaten.Betrag
  FROM Vertriebsdaten WHERE Vertriebsdaten.Verkaeufernr 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 Vertriebsdaten.Verkaeufernr = 'SP-1'

```

Funktionen

FileMaker SQL unterstützt viele Funktionen, die Sie in Ausdrücken verwenden können. Einige der Funktionen geben Zeichenfolgen zurück, einige Zahlen und einige Datumswerte.

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
CURRENT_USER	Gibt die zur Verbindungszeit angegebene Anmeldekennung zurück.	
DAYNAME	Gibt den Namen des Tages zurück, der einem angegebenen Datum entspricht.	
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 angefangen links zurück	LEFT('Mattson', 3) ergibt 'Mat'
MONTHNAME	Ergibt den Namen des Kalendermonats.	
RIGHT	Gibt die Zeichen angefangen rechts zurück	RIGHT('Mattson', 4) ergibt 'tson'
SUBSTR SUBSTRING	Gibt eine Unterzeichenfolge einer Zeichenfolge mit Parametern der Zeichenfolge, dem ersten zu extrahierenden Zeichen und der Anzahl der zu extrahierenden Zeichen zurück (optional).	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({D '25.12.2008'}) ergibt '25.12.2008'
TIME TIMEVAL	Ergibt die Tageszeit als Zeichenfolge	Um 21:49 ergibt TIME() 21:49:00
USERNAME USER	Gibt die zur Verbindungszeit angegebene Anmeldekennung zurück.	

Hinweis Die Funktion TIME() wird abgelehnt. Verwenden Sie stattdessen den SQL-Standard CURRENT_TIME.

Beispiele

```
SELECT CHR(67) + SPACE(1) + CHR(70) FROM Verkaeufere
SELECT RTRIM(' ' + Verkaeufere.Verkaeufere) AS agg FROM Verkaeufere
SELECT TRIM(SPACE(1) + Verkaeufere.Verkaeufere) AS agg FROM Verkaeufere
SELECT LTRIM(' ' + Verkaeufere.Verkaeufere) AS agg FROM Verkaeufere
SELECT UPPER(Verkaeufere.Verkaeufere) AS agg FROM Verkaeufere
SELECT LOWER(Verkaeufere.Verkaeufere) AS agg FROM Verkaeufere
SELECT LEFT(Verkaeufere.Verkaeufere, 5) AS agg FROM Verkaeufere
SELECT RIGHT(Verkaeufere.Verkaeufere, 7) AS agg FROM Verkaeufere
SELECT SUBSTR(Verkaeufere.Verkaeufere, 2, 2) +
SUBSTR(Verkaeufere.Verkaeufere, 4, 2) AS agg FROM Verkaeufere
SELECT SUBSTR(Verkaeufere.Verkaeufere, 2) + SUBSTR(Verkaeufere.Verkaeufere,
4) AS agg FROM Verkaeufere
SELECT SPACE(2) + Verkaeufere.Verkaeufere AS Verkaeufere FROM Verkaeufere
SELECT STRVAL('60506') AS agg FROM Vertriebsdaten WHERE
Vertriebsdaten.Rechnung= 1
```

Funktionen, die Zahlen zurückgeben

Funktionen, die Zahlen zurückgeben	Beschreibung	Beispiel:
ABS	Ergibt den absoluten Wert des numerischen Ausdrucks.	
ATAN	Ergibt den Arcustangens des Arguments als Winkel in Bogenmaß.	
ATAN2	Ergibt den Arcustangens der x- und y-Koordinaten als Winkel in Bogenmaß.	
B	Gibt die dezimale Entsprechung einer Binärzahl zurück.	B '1001' ergibt 9
CEIL CEILING	Ergibt den kleinsten Ganzzahlwert, der größer oder gleich dem Argument ist.	
DEG DEGREES	Ergibt die Anzahl an Grad des Arguments, das ein Winkel ist, in Bogenmaß	
DAY	Gibt den Tagesteil eines Datums zurück.	DAY({d '30.01.2010'}) ergibt 30
DAYOFWEEK	Gibt den Tag der Woche (1-7) eines Datumsausdrucks zurück.	DAYOFWEEK({d '01.05.2004'}) ergibt 7
MOD	Teilt zwei Zahlen und gibt den Rest der Division zurück.	MOD(10,3) ergibt 1
EXP	Ergibt einen Wert, der die Basis des natürlichen Logarithmus (e) hoch des Arguments ist.	
FLOOR	Ergibt den größten Ganzzahlwert, der kleiner oder gleich dem Argument ist.	
HOURL	Gibt den Stundenteil eines Werts zurück.	
INT	Gibt den ganzzahligen Teil einer Zahl zurück.	INT(6,4321) ergibt 6
LEN LENGTH	Gibt die Länge einer Zeichenfolge zurück.	LEN('ABC') ergibt 3
MONTH	Gibt den Monatsteil eines Datums zurück.	MONTH({d '30.01.2010'}) ergibt 1
LN LOG	Ergibt den natürlichen Logarithmus des Arguments.	
MAX	Gibt die größere von zwei Zahlen zurück.	MAX(66,89) ergibt 89
MIN	Gibt die kleinere von zwei Zahlen zurück.	MIN(66,89) ergibt 66
MINUTE	Gibt den Minutenteil eines Werts zurück.	
NUMVAL	Konvertiert eine Buchstabenzeichenfolge in eine Zahl; wenn die Buchstabenzeichenfolge keine gültige Zahl ist, wird 0 zurückgegeben.	NUMVAL('123') ergibt 123
PI	Gibt den konstanten Wert der mathematischen Konstante pi zurück.	
POW	Erhebt eine Zahl zu einer Potenz.	POW(7,2) ergibt 49
RADIANS	Gibt das Bogenmaß für ein Argument zurück, das in Grad ausgedrückt ist.	
ROUND	Rundet eine Zahl.	ROUND(123.456,0) ergibt 123 ROUND(123.456,2) ergibt 123.46 ROUND(123.456,-2) ergibt 100
SECOND	Gibt den Sekundenteil eines Werts zurück.	
SIGN	Ein Indikator des Vorzeichens des Arguments: -1 für negativ, 0 für 0 und 1 für positiv.	

Funktionen, die Zahlen zurückgeben	Beschreibung	Beispiel:
SIN	Gibt den Sinus des Arguments zurück.	
SQRT	Gibt die Quadratwurzel des Arguments zurück.	
TAN	Gibt den Tangens des Arguments zurück.	
VAL	Konvertiert eine Buchstabenzeichenfolge in eine Zahl; wenn die Buchstabenzeichenfolge keine gültige Zahl ist, wird 0 zurückgegeben.	VAL ('123') ergibt 123
X	Gibt die dezimale Entsprechung einer Hexadezimalzahl zurück.	X 'b9' ergibt 185
YEAR	Gibt den Jahresteil eines Datums zurück.	YEAR ({d '30.01.2010'}) ergibt 2010

Funktionen, die Datumswerte zurückgeben

Funktionen, die Datumswerte zurückgeben	Beschreibung	Beispiel:
CURDATE CURRENT_DATE	Gibt das aktuelle Systemdatum zurück.	
CURTIME CURRENT_TIME	Gibt die aktuelle Uhrzeit zurück.	
CURTIMESTAMP CURRENT_TIMESTAMP TIMESTAMPVAL	Gibt den aktuellen Zeitstempelwert zurück.	
DATE TODAY	Gibt das aktuelle Systemdatum zurück.	Wenn heute der 21.11.10 ist, ergibt DATE () 21.11.2010.
DATEVAL	Konvertiert eine Buchstabenzeichenfolge in ein Datum.	DATEVAL ('30.01.2011') ergibt 30.01.2011.

Hinweis Die Funktion DATE() wird abgelehnt. Verwenden Sie stattdessen den SQL-Standard CURRENT_DATE.

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 einst_datum > {d '30.01.2008'} UND abt = 'D101'
```

Weil AND zuerst ausgewertet wird, ruft diese Abfrage Angestellte in Abteilung A101 ab, die nach dem 30.01.08 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 einst_datum > {d '30.01.2008'}) UND abt = 'D101'
```

ruft Angestellte in Abteilung A101 ab, die entweder mehr als 40.000 Euro verdienen oder nach dem 30.01.08 eingestellt wurden.

ODBC-Katalogfunktionen

Der ODBC-Client-Treiber unterstützt die folgenden Katalogfunktionen:

- SQLTables - Kataloginformationen werden gespeichert und als einzelne Bereichsnamen gemeldet (nur Tabellenname).
- 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)
```

ABSOLUTE	COMMIT	DOUBLE	INSERT	OF
ACTION	CONNECT	DROP	INT	ON
HINZUFÜGEN	CONNECTION	ELSE	INTEGER	ONLY
ALL	CONSTRAINT	END	INTERSECT	OPEN
ALLOCATE	CONSTRAINTS	END_EXEC	INTERVAL	OPTION
ALTER	CONTINUE	ESCAPE	INTO	ORDER
AND	CONVERT	EVERY	IS	ORDER
ANY	CORRESPONDING	EXCEPT	ISOLATION	OUTER
ARE	COUNT	EXCEPTION	JOIN	OUTPUT
AS	CREATE	EXEC	KEY	OVERLAPS
ASC	CROSS	EXECUTE	LANGUAGE	PAD
ASSERTION	CURDATE	EXISTS	LAST	PART
AT	CURRENT	EXTERNAL	LEADING	PARTIAL
AUTHORIZATION	CURRENT_DATE	EXTRACT	LEFT	POSITION
AVG	CURRENT_TIME	FALSE	LENGTH	PRECISION
BEGIN	CURRENT_TIMESTAMP	FETCH	LEVEL	PREPARE
BETWEEN	CURRENT_USER	FIRST	LIKE	PRESERVE
BINARY	CURSOR	FLOAT	LOCAL	PRIMARY
BIT	CURTIME	FOR	LONGVARBINARY	PRIOR
BIT_LENGTH	CURTIMESTAMP	FOREIGN	LOWER	PRIVILEGES
BLOB	DATE	FOUND	LTRIM	PROCEDURE
BOOLEAN	DATEVAL	FROM	MATCH	PUBLIC
BOTH	DAY	FULL	MAX	READ
BY	DAYNAME	GET	MIN	REAL
CASCADE	DAYOFWEEK	GLOBAL	MINUTE	REFERENCES
CASCADED	DEALLOCATE	GO	MODULE	RELATIVE
CASE	DEC	GOTO	MONTH	RESTRICT
CAST	DECIMAL	GRANT	MONTHNAME	REVOKE
CATALOG	DECLARE	GROUP	NAMES	RIGHT
CHAR	DEFAULT	HAVING	NATIONAL	ROLLBACK
CHARACTER	DEFERRABLE	HOURL	NATURAL	ROUND
CHARACTER_LENGTH	DEFERRED	IDENTITY	NCHAR	ROWID
CHAR_LENGTH	DELETE	IMMEDIATE	Next	ROW
CHECK	DESC	IN	NO	RTRIM
CHR	DESCRIBE	INDEX	NOT	SCHEMA
CLOSE	DESCRIPTOR	INDICATOR	NULL	SCROLL
COALESCE	DIAGNOSTICS	INITIALLY	NULLIF	SECOND
COLLATE	DISCONNECT	INNER	NUMERIC	SECTION
COLLATION	DISTINCT	INPUT	NUMVAL	SELECT
COLUMN	DOMAIN	INSENSITIVE	OCTET_LENGTH	SESSION

SESSION_USER	VALUE
SET	VALUES
SIZE	VARBINARY
SMALLINT	VARCHAR
SOME	VARYING
SPACE	VIEW
SQL	WHEN
SQLCODE	WHENEVER
SQLERROR	WHERE
SQLSTATE	WITH
STRVAL	WORK
SUBSTRING	WRITE
SUM	YEAR
SYSTEM_USER	ZONE
TABLE	
TEMPORARY	
THEN	
TIME	
TIMESTAMP	
TIMESTAMPVAL	
TIMEVAL	
TIMEZONE_HOUR	
TIMEZONE_MINUTE	
TO	
TODAY	
TRAILING	
TRANSACTION	
TRANSLATE	
TRANSLATION	
TRIM	
TRUE	
UNION	
UNIQUE	
UNKNOWN	
UPDATE	
UPPER	
USAGE	
USER	
USERNAME	
USING	

Kapitel 8

Referenzinformationen

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.

Hinweis FileMaker-Wiederholfelder werden wie Arrays unterstützt. Beispiele:

```
INSERT INTO mytable(repField[3]) VALUES ('Das ist Wdhg 3')
SELECT repField[1], repField[2] FROM mytable
```

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.)

FileMaker-Feldtyp	JDBC SQL-Typ
Text	java.sql.Types.VARCHAR
Zahl	java.sql.Types.DOUBLE

FileMaker-Feldtyp	JDBC SQL-Typ
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.

ODBC- und JDBC-Fehlermeldungen

Im Folgenden sind die grundlegenden Fehlermeldungen aufgeführt, die beim Arbeiten mit FileMaker und ODBC/JDBC auftreten können.

ODBC-Fehlermeldungen

Fehlermeldungen werden ausgegeben wegen:

- ODBC-Treiberfehlern
- FileMaker- und FileMaker xDBC Listener-Fehler

FileMaker ODBC-Fehlermeldungen

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

[FileMaker] [FileMaker ODBC] Meldung

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

[FileMaker] [FileMaker ODBC] Ungültiger Benutzername/Passwort

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

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

- JDBC-Treiberfehlern
- FileMaker- und FileMaker xDBC Listener-Fehler

FileMaker JDBC-Fehlermeldungen

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

[FileMaker] [FileMaker JDBC] Meldung

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

[FileMaker] [FileMaker JDBC] Ungültiger Benutzername/Passwort

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

ABS, Funktion 55
ALL, Operator 51
ALTER TABLE (SQL-Anweisung) 46
Anforderungen für die Installation 15, 29
ANY, Operator 51
ARRAY-Datentyp 32
ATAN, Funktion 55
ATAN2, Funktion 55
Ausdrücke in SQL 49
Ausschnitte 37
Automatisch erstellte Schlüssel 32

B

B, Funktion 55
BETWEEN, Operator 51
Bilddateien in Medienfeldern 43
Binärdaten
 Verwenden in SELECT 42
bitmap-Dateien in Medienfeldern 43
BLOB-Datentyp
 Verwenden in SELECT 42
 Verwendung in CREATE TABLE 46
Boolescher Datentyp 32

C

CAST-Funktion 43, 61
CEIL, Funktion 55
CEILING, Funktion 55
CHR, Funktion 53
Client-Anwendung, Verwenden von FileMaker als 7
CLOB-Datentyp 32
CREATE INDEX (SQL-Anweisung) 47
CREATE TABLE (SQL-Anweisung) 46
CURDATE, Funktion 56
CURRENT USER, Funktion 53
CURRENT_DATE, Funktion 56
CURRENT_TIME, Funktion 56
CURRENT_TIMESTAMP, Funktion 56
CURRENT_USER, Funktion 53
Cursor
 In JDBC 32
 In ODBC 41

CURTIME, Funktion 56
CURTIMESTAMP, Funktion 56

D

DATALINK-Datentyp 32
DATE, Funktion 56
Dateien
 Einrichten des Zugriffs auf 23
 Organisation auf einem Computer 9
 Verwenden in Medienfeldern 43
Datenbank, DSN 25
Datenbankverbindungen, Anzahl an unterstützten 9
Datenquelle
 ein Datenquellenname für jede FileMaker-Datenbankdatei 10
 eine gemeinsam verwendete FileMaker-Datenbankdatei deaktivieren 10
 für Zugriff über JDBC konfigurieren 34
 Konfigurieren für Zugriff über ODBC (Mac OS) 26
 Konfigurieren für Zugriff über ODBC (Windows) 24
 Überprüfen des Zugriffs über ODBC (Windows) 25
 Zugriff über JDBC überprüfen 35
 Zugriffsüberprüfen über ODBC (Mac OS) 27
Datenquellennamen
 Einer pro Datei 10
 Erstellen (MacOS) 26
 Erstellen (Windows) 24
Datenquellennamen *siehe* Datenquellennamen
Datentyp
 JDBC-Client-Treiber 61
 ODBC-Client-Treiber 61
Datentypen zuordnen
 JDBC-Client-Treiber 61
 ODBC-Client-Treiber 61
DATEVAL, Funktion 56
Datumsformate 49
Datumsoperatoren in SQL-Ausdrücken 50
DAY, Funktion 55
DAYNAME, Funktion 53
DAYOFWEEK, Funktion 55
DEG, Funktion 55
DEGREES, Funktion 55
DELETE (SQL-Anweisung) 44

DISTINCT, Operator 38
 DROP INDEX (SQL-Anweisung) 47

E

eine gemeinsam verwendete FileMaker-Datenbankdatei deaktivieren 10
 Erweiterte Zugriffsrechte 23
 EXISTS, Operator 51
 EXP, Funktion 55
 Exponentialschreibweise in SQL-Ausdrücken 50

F

Fehlermeldungsformate 62
 Felder
 Zuordnen zu JDBC 61
 Zuordnung zu ODBC 61
 Feldnamen in SQL-Ausdrücken 49
 FileMaker-Produkte 9
 FLOOR, Funktion 55
 FOR UPDATE (SQL-Klausel) 41
 FROM (SQL-Klausel) 39
 FULL OUTER JOIN 39
 Funktionen in SQL-Ausdrücken 53

G

GetAs-Funktion 43, 61
 GROUP BY (SQL-Klausel) 40

H

Haltbarer Cursor 32
 HAVING (SQL-Klausel) 40
 Host, DSN 24, 26
 HOUR, Funktion 55

I

IN, Operator 51
 INNER JOIN 39
 INSERT (SQL-Anweisung) 44
 Installationsanforderungen 15, 29
 INT, Funktion 55
 IS NOT NULL, Operator 51
 IS NULL, Operator 51

J

Java Development Kit (JDK) 32
 Java-Version 29

JDBC

Beschreibung 31
 Client-Treiber, beschrieben 32
 Fehlermeldungen 62
 Überblick der Verwendung 7

JDBC SPI 32

JDBC-Client-Treiber

Ausschnitte 37
 beim JDBC-Treiber-Manager registrieren 33
 Datentypen zuordnen 61
 JDBC URL angeben 32
 Metadaten-Funktionen 57
 Treiberklasse und Haupteinstieg 32
 Unicode-Unterstützung 37
 Zugriff überprüfen 35

Join 39

K

Katalogfunktionen für ODBC 57
 Konfigurieren einer FileMaker-Datenquelle über JDBC 34
 Über ODBC (Mac OS) 26
 Über ODBC (Windows) 24
 Konstanten in SQL-Ausdrücken 49
 Konten und Zugriffsrechte 23

L

Leere Werte in Spalten 44
 Leere Zeichenfolge
 Verwenden in SELECT 42
 Leerzeichen 50
 LEFT JOIN 39
 LEFT OUTER JOIN 39
 LEFT, Funktion 53
 LEN, Funktion 55
 LIKE, Operator 51
 Literale in SQL-Ausdrücken 49
 LN, Funktion 55
 LOG, Funktion 55
 Logische Operatoren in SQL-Ausdrücken 52
 LOWER, Funktion 53
 LTRIM, Funktion 53

M

Mac OS

Erstellen eines Datenquellennamens 26
 JDBC-Client-Treiber 29
 ODBC-Client-Treiber-Anforderungen 15

Überprüfen des ODBC-Zugriffs 27

MAX, Funktion 55

Medienfeld

- JDBC-Datentypzuordnung 62
- Mit INSERT-Anweisung 44
- Mit SELECT-Anweisung 42
- Mit UPDATE-Anweisung 45
- ODBC-Datentypzuordnung 61

Metadaten-Funktionen für JDBC 57

MIN, Funktion 55

MINUTE, Funktion 55

MOD, Funktion 55

MONTH, Funktion 55

MONTHNAME, Funktion 53

N

Netzwerkanforderungen 10

NICHT-Operator 52

NOT IN, Operator 51

NOT LIKE, Operator 51

NOT NULL (SQL-Klausel) 46

Nullwert 44, 61

Numerische Operatoren in SQL-Ausdrücken 50

NUMVAL, Funktion 55

O

ODBC

- Beschreibung 21
- Fehlermeldungen 62
- Standards, Einhaltung 37
- Überblick der Verwendung 7
- Wiederholfelder 61

ODBC-Administrator (Mac OS) 27

ODBC-Client-Treiber

- Ausschnitte 37
- Datentypen zuordnen 61
- Katalogfunktionen 57
- Unicode-Unterstützung 37
- Zugriff überprüfen (Mac OS) 27
- Zugriff überprüfen (Windows) 25

ODBC-Datenquellenadministrator (Windows) 26

ODER-Operator 52

Operatorpriorität bei SQL-Ausdrücken 56

ORDER BY (SQL-Klausel) 41

OUTER JOIN 39

P

Passwort

Mit JDBC 34

Mit ODBC 25, 27

PI, Funktion 55

Port, für JDBC angeben 30

Positioned Updates und Deletes 41

POW, Funktion 55

PREVENT INDEX CREATION 48

Q

QuickTime-Dateien in Medienfeldern 43

R

RADIANS, Funktion 55

Rapid Application Development (RAD)-Tools 31

REF-Datentyp 32

Registrieren des JDBC-Client-Treibers 33

Relationale Operatoren in SQL-Ausdrücken 51

Remote-Zugriff 9

Reservierte SQL-Schlüsselwörter 57

RIGHT JOIN 39

RIGHT OUTER JOIN 39

RIGHT, Funktion 53

ROUND, Funktion 55

RTRIM, Funktion 53

S

Schlüsselwörter, reservierte SQL- 57

SECOND, Funktion 55

SELECT (SQL-Anweisung) 37

- Binärdaten 42
- BLOB-Datentyp 42
- Leere Zeichenfolge 42

Server-Datenquelle 27

Sharing, Einrichten von ODBC/JDBC- 23

Sicherungspunkt-Unterstützung 32

SIGN, Funktion 55

SIN, Funktion 56

SPACE, Funktion 53

Spaltenaliasse 38

Spaltennamen 23

SQL_C_WCHAR, Datentyp 37

SQL-92 37

SQL-Anweisungen

- ALTER TABLE 46
- CREATE INDEX 47
- CREATE TABLE 46
- DELETE 44

DROP INDEX 47
 INSERT 44
 Reservierte Schlüsselwörter 57
 SELECT 37
 unterstützt von Client-Treibern 37
 UPDATE 45
 SQL-Ausdrücke 49
 Datumsoperatoren 50
 Exponentialschreibweise 50
 Feldnamen 49
 Funktionen 53
 Konstanten 49
 Literale 49
 logische Operatoren 52
 numerische Operatoren 50
 Priorität der Operatoren 56
 relationale Operatoren 51
 Zeichenoperatoren 50
 SQLExceptions 62
 SQL-Standards, Einhaltung 37
 SQL-Statistikfunktionen 48
 SQRT, Funktion 56
 Standards, Einhaltung 37
 Statistikfunktionen in SQL 48
 STRVAL, Funktion 53
 SUBSTR, Funktion 53
 SUBSTRING, Funktion 53
 Syntaxfehler 57
 Systemanforderungen 15, 29

T

Tabellenaliasse 38, 39
 TAN, Funktion 56
 Testen des Zugriffs
 JDBC-Client-Treiber 35
 ODBC-Client-Treiber (MacOS) 27
 ODBC-Client-Treiber (Windows) 25
 TIME, Funktion 53
 TIMESTAMPVAL, Funktion 56
 TIMEVAL, Funktion 53
 TODAY, Funktion 56
 Treiber
 Deinstallation alter 10
 Treibereigenschaften
 JDBC-Client-Treiber 34
 ODBC-Client-Treiber (MacOS) 26
 ODBC-Client-Treiber (Windows) 24
 TRIM, Funktion 53

U

Übersicht
 Einrichten von Zugriffsrechten und Sharing 23
 Verwenden von ODBC und JDBC mit
 FileMaker 7
 UND-Operator 52
 Unicode-Unterstützung 37
 UNION (SQL-Operator) 40
 Unterabfragen 44
 UPDATE (SQL-Anweisung) 45
 UPPER, Funktion 53
 URL (Uniform Resource Locator) für den JDBC-
 Client-Treiber 32
 USERNAME, Funktion 53

V

VAL, Funktion 56
 VALUES (SQL-Klausel) 44
 Verbindungen, Datenbank 9

W

WHERE (SQL-Klausel) 39
 Wiederholfelder 61
 Windows
 Erstellen eines Datenquellennamens 24
 JDBC-Client-Treiber 29
 ODBC-Client-Treiber-Anforderungen 15
 Überprüfen des ODBC-Zugriffs 25

X

X, Funktion 56

Y

YEAR, Funktion 56

Z

Zeichenfolge, Funktionen 53
 Zeichenoperatoren in SQL-Ausdrücken 50
 Zeitformate 49
 Zeitstempelformate 49
 Zugriff über ODBC/JDBC, erweitertes
 Zugriffsrecht 23
 Zugriff überprüfen
 JDBC-Client-Treiber 35
 ODBC-Client-Treiber (MacOS) 27
 ODBC-Client-Treiber (Windows) 25
 Zugriffsrechte, erweitert 23