

FileMaker 7

ODBC- och JDBC-handbok för utvecklare



© 2004, FileMaker, Inc. Med ensamrätt.

FileMaker, Inc.

5201 Patrick Henry Drive

Santa Clara, Kalifornien 95054, USA

FileMaker är ett varumärke som tillhör FileMaker, Inc., registrerat i USA och andra länder, och ScriptMaker och logotypen för mappen är varumärken som tillhör FileMaker, Inc.

FileMakers dokumentation skyddas av lagen om upphovsrätt och det är därför inte tillåtet att mångfaldiga eller distribuera detta dokument utan FileMakers skriftliga medgivande. Dokumentationen får endast användas tillsammans med ett licensierat exemplar av FileMaker-programmet.

Samtliga personer och företag som förekommer i exempel är fiktiva och eventuella likheter med verkliga personer och företag är fullständigt oavsiktliga.

En lista över medverkande visas i dokumentet Tillskrivning som medföljer den här programvaran.

Mer information finns på webbplatsen www.filemaker.com.

Utgåva: 01

Innehåll

Kapitel 1	
<i>Introduktion</i>	5
Om denna handbok	5
Använda ODBC och JDBC tillsammans med FileMaker	5
Använda en FileMaker-databasfil som en datakälla	6
Nätverkskrav	6
Använda FileMaker Pro som ett klientprogram	6
Uppdatera filer från tidigare versioner	6
Om du tidigare delade en FileMaker-databasfil som en datakälla	7
Om du tidigare använde FileMaker Pro som en klient för åtkomst till en datakälla	7
Kapitel 2	
<i>Delat FileMaker-data med hjälp av ODBC</i>	9
Om ODBC	9
Använda ODBC-klientdrivrutinen	10
Installera ODBC-klientdrivrutinen	10
Ange ODBC-klientdrivrutinsegenskaper för ett FileMaker-datakällnamn	10
Kontrollera åtkomst via ODBC/JDBC	11
Ansluta till en FileMaker-databasfil	12
Kapitel 3	
<i>Delat FileMaker-data med hjälp av JDBC</i>	13
Om JDBC	13
Använda JDBC-klientdrivrutinen	13
Installera JDBC-klientdrivrutinen	13
Om JDBC-klientdrivrutinen	14
Använda en JDBC-URL för att ansluta till en databas	14
Specificera drivrutinens egenskaper i URL-delnamnet	15
Kontrollera åtkomst via JDBC	16
Kapitel 4	
<i>Standarder som stöds</i>	17
Stöd för Unicode-tecken	17
SQL-uttryck och -instruktioner	17
SELECT-uttryck	17
DELETE-uttryck	19
INSERT-uttryck	19
UPDATE-uttryck	20
CREATE TABLE-uttryck	21
ALTER TABLE-uttryck	22
CREATE INDEX-uttryck	22

4 FileMaker ODBC- och JDBC-handbok för utvecklare

DROP INDEX-uttryck	22
FROM-instruktion	23
WHERE-instruktion	23
GROUP BY-instruktion	23
HAVING-instruktion	24
UNION-operator	24
ORDER BY-instruktion	24
FOR UPDATE-instruktion	25
Statistikfunktioner i SQL	25
SQL-uttryck	26
Fältnamn	26
Konstanter och ordagranna strängar	26
Exponentiell/matematisk notation	27
Numeriska operatorer	27
Teckenoperatorer	27
Datumoperatorer	28
Relationsoperatorer	28
Logiska operatorer	29
Funktioner	30
Prioritetsordning för operatorer	32
ODBC-katalogfunktioner	32
Metadatafunktioner för JDBC	33
Bilaga A	
<i>Mappa FileMaker-fält till ODBC-datatyper</i>	35
Bilaga B	
<i>Mappa FileMaker-fält till JDBC-datatyper</i>	37
Bilaga C	
<i>ODBC- och JDBC-felmeddelanden</i>	39
ODBC-felmeddelanden	39
Felmeddelanden som rör ODBC-drivrutinen	39
Felmeddelanden som rör ODBC-drivrutinshanteraren	39
Felmeddelanden som rör SequeLink-klienten	39
Felmeddelanden som rör SequeLink-servern	39
Felmeddelanden som rör datakällan	40
JDBC-felmeddelanden	40
Felmeddelanden som rör JDBC-drivrutinen	40
Felmeddelanden som rör SequeLink-servern	40
Felmeddelanden som rör datakällan	40
<i>Index</i>	41

Kapitel 1

Introduktion

Välkommen till *FileMaker® ODBC- och JDBC-handbok för utvecklare*. Den här handboken innehåller information om hur du delar FileMaker-data med andra program genom att använda ODBC (Open Database Connectivity) och JDBC (Java Database Connectivity). Boken beskriver också hur ODBC- och JDBC-klientdrivrutinerna, när de används tillsammans med FileMaker Pro och FileMaker Server, stöder branschstandarderna för ODBC, JDBC och SQL (Structured Query Language).

Du kan använda FileMaker Pro, FileMaker Server eller FileMaker Developer för att skapa och testa databaslösningar. Du kan därefter dela din FileMaker-databaslösning som datakälla med ODBC- och JDBC-kompatibla program eller använda andra ODBC-datakällor genom att använda din FileMaker-databaslösning som ett klientprogram.

Om denna handbok

- Den här handboken innehåller endast information om hur du använder och delar data genom att använda ODBC och JDBC tillsammans med FileMaker Pro 7, FileMaker Developer 7 och FileMaker Server 7. Information om hur du använder ODBC och JDBC med tidigare versioner av FileMaker Pro kan du hämta från www.filemaker.com/intl.
- I den här handboken förutsätts att du känner till grunderna i hur du använder ODBC/JDBC och skapar SQL-frågor. Mer information om dessa ämnen finns i böcker från olika företag.
- Stegvisa instruktioner om FileMaker Pro-funktioner, bland annat om hur du använder datakällor via ODBC och ett exempel på ODBC-import, finns i FileMaker Pro Hjälp.
- I FileMaker Pro-dokumentationen används uttrycket *webbpublicering*, som beskriver databaser som användare kan komma åt via Internet eller ett intranät med hjälp av en webbläsare.
- I denna handbok avser termen "FileMaker Pro" både FileMaker Pro och FileMaker Developer, utom när specifika FileMaker Developer-funktioner beskrivs.
- I det här dokumentet avser termen "FileMaker Server" FileMaker Server 7 Advanced, som stöder delning av FileMaker-databasfiler med andra program som kan hantera ODBC och JDBC.

Viktigt! Du kan hämta PDF-filer med FileMaker 7-dokumentation från www.filemaker.com/intl. Uppdateringar till den här handboken finns också att hämta från den webbadressen.

Använda ODBC och JDBC tillsammans med FileMaker

ODBC och JDBC är programmeringsgränssnitt (API:er). Dessa API:er ger klientprogrammen ett gemensamt språk så att de kan samverka med olika datakällor och databastjänster, bland annat FileMaker Pro och FileMaker Server.

Alla program som har stöd för ODBC och JDBC kan tolka en grundläggande delmängd av SQL-uttryck. När du arbetar med SQL kan du använda andra program (som kalkylblad, ordbehandlingsprogram och rapportverktyg) för att visa, analysera och ändra FileMaker-data.

SQL-uttrycken överförs via ODBC- och JDBC-gränssnitten till datakällans FileMaker-värd. Du kan ställa frågor som `SELECT förnamn, efternamn FROM kunder WHERE ort='Paris'` och göra uppdateringar, till exempel skapa en ny post med `INSERT INTO kunder (förnamn, efternamn) VALUES ('Johanna', 'Svensson')`.

Använda en FileMaker-databasfil som en datakälla

Du kan använda FileMaker Server som värd för en FileMaker-databasfil som datakälla och dela dina data med andra program genom att använda ODBC och JDBC (endast Windows). FileMaker Server stöder lokal åtkomst (från samma dator) och fjärråtkomst (både från programsystem (s.k. middleware) som bland annat webbservrar och, för fjärrklientåtkomst, från skrivbordsprogram). Om du är värd för en FileMaker-databasfil med FileMaker Server gör du klienterna tillgängliga för fjärranvändare. (Installationsprogram och instruktioner finns på installationsskivan i mappen \xDBC.)

Du kan använda FileMaker Pro som värd för en FileMaker-databasfil som datakälla och dela dina data med klientprogram genom att använda ODBC och JDBC (endast Windows). FileMaker Pro godkänner upp till fem anslutningar och programmet stöder endast lokal åtkomst (från samma dator).

Ibland använder klientprogram andra termer för åtkomst till en datakälla. Många program har menyalternativ med namn som t.ex. Hämta externa data eller SQL-fråga. Mer information finns i dokumentationen eller hjälpen för programmet.

I Kapitel 4, ”Standarder som stöds”, beskrivs vilka SQL-uttryck som ODBC- och JDBC-klienterna stöder när du använder FileMaker Pro och FileMaker Server. Vidare information om hur du skapar SQL-frågor finns i allmänna böcker om SQL.

Viktigt! Om du inaktiverar ODBC/JDBC-delning när det har varit aktiverat, blir en datakälla som finns på FileMaker Server eller FileMaker Pro genast otillgänglig. Databasadministratören kan inte meddela ODBC- och JDBC-klientprogram om datakällans tillgänglighet (administratören kan endast kommunicera med FileMaker-databasklienter). Inga fel rapporteras och klientprogrammet måste meddela användarna att datakällan inte är tillgänglig och att transaktioner inte kan slutföras. Om ett klientprogram försöker ansluta till en otillgänglig FileMaker-databasfil, visas ett meddelande om att anslutningen misslyckades.

I handboken *Installation av FileMaker-drivrutiner för ODBC- och JDBC-klienter* finns information om hur du installerar de drivrutinsfiler som behövs för att du ska kunna använda en FileMaker-datakälla.

Nätverkskrav

Du behöver ett TCP/IP-nätverk om du använder FileMaker Server 7 som värd för en FileMaker-databasfil som datakälla i ett nätverk. FileMaker Pro 7 stöder endast lokal åtkomst (från samma dator).

Använda FileMaker Pro som ett klientprogram

FileMaker Pro kan användas som ett ODBC-klientprogram, som samverkar med datakällor på samma dator eller i ett nätverk (Windows och Mac OS).

Uppdatera filer från tidigare versioner

I FileMaker Pro 7 kan filer som skapats i tidigare FileMaker-versioner öppnas. Läs mer i *Konvertera FileMaker-databaser från tidigare versioner*.

FileMaker Server 7 kan endast fungera som värd för FileMaker Pro 7-databasfiler och kan inte vara värd för eller konvertera tidigare versioner av FileMaker Pro-databasfiler.

Om du tidigare delade en FileMaker-databasfil som en datakälla

Om du använde LDAC (Local Data Access Companion) eller RDAC (Remote Data Access Companion) för att dela en databasfil i tidigare versioner, måste du göra två ändringar. Först måste du koppla användare till ett konto som har den utökade behörigheten Åtkomst via ODBC/JDBC (som en del av kontots behörighetsuppsättning). Därefter måste du aktivera din FileMaker Server- eller FileMaker Pro-värd för ODBC/JDBC-delning (i FileMaker Pro väljer du Redigera > Dela databas > ODBC/JDBC för att aktivera delning).

I FileMaker Pro Hjälp finns utförlig information om delning via ODBC/JDBC och hur du konfigurerar konton och behörighetsuppsättningar.

Obs! I tidigare versioner av FileMaker Server och FileMaker Pro kunde du bara skapa ett DSN-namn (Data Source Name, datakällnamn) för värdprogrammet när du konfigurerade en drivrutin för åtkomst via ODBC. Detta enda DSN-namn möjliggjorde åtkomst till FileMaker-databasfilerna som datakälla. Nu måste du skapa ett DSN-namn för varje enskild FileMaker-databasfil som du vill använda som datakälla. Om du redan har konfigurerat åtkomst via ett DSN-namn som tillåter att tabeller används av flera FileMaker-databasfiler, måste du slå samman dessa tabeller till en enda databasfil (eller skapa fler DSN-namn).

Om du tidigare använde FileMaker Pro som en klient för åtkomst till en datakälla

De drivrutiner du konfigurerade för åtkomst till datakällan bör fortfarande fungera. För att kunna utnyttja alla fördelar med de nya drivrutinerna bör du dock konfigurera om drivrutinerna.

FileMaker innehåller klientdrivrutiner för följande ODBC-datakällor:

- Text (Windows och Mac OS)
- Oracle 9i version 2 (Windows och Mac OS)
- SQL Server 2000 (Mac OS)
- FileMaker Pro 7 (Windows)

I Windows ingår dessutom klientdrivrutiner, till exempel SQL Server-drivrutinen, för operativsystemet.

Om du vill samverka med en datakälla som saknar en drivrutin i listan kan du skaffa ODBC-drivrutiner från många fristående leverantörer.

I FileMaker Pro Hjälp finns utförlig information om hur du använder FileMaker Pro för att få åtkomst till andra datakällor via ODBC.

Kapitel 2

Dela FileMaker-data med hjälp av ODBC

Använd ODBC-klientdrivrutinen, DataDirect 32-BIT Sequelink 5.4, när du vill ansluta till en FileMaker-datakälla från ett annat program (Windows). Det program som använder ODBC-klientdrivrutinen har direktåtkomst till informationen i en FileMaker-databasfil.

Du kan också använda FileMaker Pro som ett klientprogram och på så sätt använda poster från en annan datakälla via SQL (Windows och Mac OS). I FileMaker Pro Hjälp finns utförlig information om vilka drivrutiner från FileMaker som ingår och hur du får åtkomst till en datakälla med hjälp av drivrutinerna.

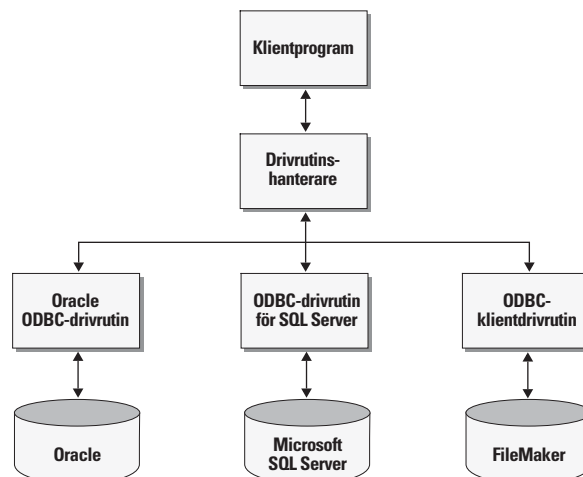
Om ODBC

ODBC är ett programgränssnitt (API) som ger program åtkomst till data från många olika DBMS-system (system för databashantering). ODBC ger klientprogrammen ett gemensamt språk så att de kan samverka med datakällor och databastjänster.

Alla program som har stöd för ODBC kan tolka en gemensam delmängd av SQL-uttryck. När du arbetar med SQL kan du använda andra program (som kalkylblad, ordbehandlingsprogram och rapportverktyg) för att visa, analysera och ändra FileMaker-data. I kapitel 4, "Standarder som stöds", finns information om vilka SQL-uttryck och SQL-funktioner ODBC-klientdrivrutinen stöder.

Via ODBC-klientdrivrutinen kan programmet kommunicera direkt med en FileMaker-databasfil. Dina SQL-instruktioner sänds till FileMaker-värden för databasfilen och resultatet av instruktionerna sänds tillbaka till dig. Om du använder FileMaker Server som värd för en FileMaker-databasfil som datakälla, kan databasen ligga på en annan dator (serverdatorn) som är ansluten till nätverket medan klientprogrammet finns på din dator (klientdatorn). Det här kallas för klient-/serverkonfiguration.

ODBC-komponenter



Använda ODBC-klientdrivrutinen

Du kan använda ODBC-klientdrivrutinen med alla ODBC-kompatibla Windows-program. Genom att dela din FileMaker-databasfil som en datakälla kan du göra följande:

- Koppla utskrifter med Microsoft Word.
- Skapa diagram med Microsoft Excel.
- Flytta FileMaker-data till ett DBMS-system, t.ex. Microsoft SQL Server.
- Göra ytterligare analyser av dina FileMaker-data med fråge- och rapportverktyg för att skapa diagram, konstruera ad hoc-frågor och utföra djupanalyser.
- Skapa ett Microsoft Visual Basic-program som delar information med FileMaker Pro.

Om du vill dela en FileMaker-databasfil som datakälla använder du FileMaker Pro för att definiera vilka konton som ska ha åtkomst till databasfilen. Styr sedan åtkomsten till databasfilen genom att tilldela behörighetsuppsättningar till kontona, inklusive den utökade behörigheten Åtkomst via ODBC/JDBC. Slutligen aktiverar du det FileMaker Server- eller FileMaker Pro-värdprogram som du vill använda för att dela data via ODBC/JDBC.

Mer information finns i FileMaker Pro eller FileMaker Server Hjälp.

Viktigt! ODBC-klientdrivrutinen ersätter den ODBC-drivrutin för FileMaker Pro som medföljde tidigare versioner av FileMaker. Om du har konfigurerat åtkomst till en FileMaker-datakälla med en äldre drivrutin, måste du definiera om åtkomsten genom att använda och konfigurera den nya drivrutinen.

Installera ODBC-klientdrivrutinen

ODBC-klientdrivrutinen är tillgänglig via ett separat installationsprogram på installationsskivan i mappen \xDBC\ODBC Client Driver Installer. I PDF-filen *Installation av FileMaker-drivrutiner för ODBC- och JDBC-klienter* i mappen \xDBC\Elektronisk dokumentation på installationsskivan finns mer information.

Ange ODBC-klientdrivrutinsegenskaper för ett FileMaker-datakällnamn

Skapa ett DSN-namn (datakällnamn) för varje enskild FileMaker-databasfil som du vill ha åtkomst till som datakälla (i tidigare versioner av FileMaker Server och FileMaker Pro skapade du bara ett DSN-namn för alla FileMaker-databasfiler som programmet var värd för). DSN-namnet identifierar ODBC-drivrutinen för FileMaker (DataDirect 32-BIT SequeLink 5.4), värdprogrammet (FileMaker Server eller FileMaker Pro) och den FileMaker-databasfil som du använder som datakälla.

Så här konfigurerar eller ändrar du egenskaperna för ODBC-klientdrivrutiner:

1. Öppna ODBC - Administrera datakälla på Kontrollpanelen.

Välj Start-menyn > Inställningar > Kontrollpanelen > Administrationsverktyg > Datakällor (ODBC).

2. Klicka på fliken System-DSN.

Om du har konfigurerat datakällorna som ett användar-DSN eller fil-DSN klickar du på motsvarande flik.

3. Klicka på Lägg till.

Om du ändrar egenskaperna för en befintlig datakälla, markerar du datakällan, klickar på Konfigurera och går vidare till steg 6.

4. Välj DataDirect 32-BIT SequeLink 5.4 i listan över drivrutiner.

5. Klicka på Slutför.

Dialogrutan ODBC - inställningar för DataDirect SequeLink visas, med fliken Diverse vald.

6. Under Namn på datakälla skriver du ett namn som är meningsfullt för andra som använder FileMaker-datakällan.

Du kan ange ytterligare en beskrivning, om du vill.

Kontrollera att rutan Använd LDAP är avmarkerad. ODBC-klientdrivrutinen innehåller inga DLL-översättningsfiler (så även om du klickar på Översätt kommer du inte att kunna konfigurera en ODBC-översättare).

7. Under SequeLink-servervärd anger du sökvägen till datakällan.

Om du ansluter till en FileMaker-databas som FileMaker Pro är värd för på den lokala datorn, skriver du localhost (eller 127.0.0.1).

Om du ansluter till en FileMaker-databas som FileMaker Server är värd för över ett nätverk, skriver du IP-adressen till FileMaker Server.

8. Under SequeLink-serverport skriver du 2399.

9. Under Serverdatakälla skriver du filnamnet på den FileMaker-databasfil du använder som en datakälla (skriv inte filändelsen).

Om du har aktiverat delning via ODBC/JDBC i FileMaker Pro- eller FileMaker Server-värdprogrammet visas i listrutan till höger om Serverdatakälla namnen på de FileMaker-databasfiler som är öppna och som du kan välja bland.

10. Spara informationen genom att klicka på OK.

Om du delar en annan FileMaker-databasfil går du tillbaka till steg 3 och konfigurerar databasfilen som datakälla.

11. Klicka på OK när du vill stänga dialogrutan ODBC - Administrera datakälla.

Viktigt! När du konfigurerar en FileMaker-datakälla måste du ange 2399 som serverport för SequeLink.

Kontrollera åtkomst via ODBC/JDBC

Så här kontrollerar du att ODBC-klientdrivrutinen är korrekt konfigurerad för åtkomst till FileMaker-datakällan:

1. Öppna ODBC - Administrera datakälla på Kontrollpanelen.

Välj Start-menyn > Inställningar > Kontrollpanelen > Administrationsverktyg > Datakällor (ODBC).

2. Klicka på fliken System-DSN.

Om du har konfigurerat datakällorna som ett användar-DSN eller fil-DSN klickar du på motsvarande flik.

3. Välj den FileMaker-datakälla som du konfigurerade.

Namnet på datakällan som du ursprungligen angav visas under Namn och DataDirect 32-BIT SequeLink 5.4 visas som Drivrutin.

4. Klicka på Konfigurera.

Dialogrutan ODBC - inställningar för DataDirect SequeLink visas.

5. Klicka på Testa anslutning.

Du uppmanas att ange FileMaker-kontonamnet (i Databasanvändarnamn) och FileMaker-lösenordet (i Databaslösenord).

Om anslutningen fungerar som den ska visas meddelandet *Anslutningen har upprättats*. Om anslutningen misslyckas:

- Kontrollera att FileMaker-databasfilen har en värd och är tillgänglig.
- Uppdatera eller korrigera anslutningsinformationen.
- Se till att FileMaker-kontot använder en behörighetsuppsättning med den utökade behörigheten *Åtkomst via ODBC/JDBC*.
- Kontrollera att FileMaker Pro- eller FileMaker Server-värdprogrammet har konfigurerats för delning via ODBC/JDBC.

Ansluta till en FileMaker-databasfil

Från ett ODBC-kompatibelt program kan du skapa SQL-frågor för åtkomst till en FileMaker-databasfil. ODBC-klientdrivrutinen måste vara installerad på den dator som genererar SQL-frågan.

Så här ansluter du:

1. I FileMaker Pro granskar du de behörighetsuppsättningar du har angett för konton som ska ha åtkomst till databasfilen.

Konton som ska ha åtkomst måste ha en behörighetsuppsättning med den utökade behörigheten *Åtkomst via ODBC/JDBC*.

2. Aktivera FileMaker Server- eller FileMaker Pro-värdprogrammet som du vill använda för att dela data via ODBC/JDBC.

FileMaker Server (via Administration av FileMaker Server): Välj *Åtgärd > Egenskaper > fliken Klienter*, klicka på *Aktivera klienttjänster* och markera *Tillåt ODBC- och JDBC-anslutningar*.

FileMaker Pro: Välj *Redigera > Dela databas > ODBC/JDBC* och aktivera delning för filen.

3. Kontrollera att FileMaker-databasfilen du vill komma åt har en värd och är tillgänglig.
4. Anslut till FileMaker-datakällan.

Ibland använder klientprogram andra termer för åtkomst till en datakälla via ODBC. Många program har menyalternativ med namn som t.ex. *Hämta externa data* eller *SQL-fråga*. Dokumentationen eller hjälpen som medföljer programmet innehåller mer information.

5. Skapa och utför en SQL-fråga i klientprogrammet.

Varje FileMaker-databasfil som är öppen och konfigurerad för åtkomst är en separat datakälla (du skapar ett DSN-namn för varje enskild FileMaker-databasfil som du vill kunna använda som datakälla).

Varje databas kan ha en eller fler tabeller. FileMaker-fälten motsvaras av kolumner. Det fullständiga fältnamnet, inklusive eventuella icke-alfanumeriska tecken, visas som kolumnnamnet.

Obs! I Windows kan Microsoft Access importera högst 32 fält på en och samma gång via ODBC från en FileMaker-databasfil. Om databasfilen har mer än 32 fält importerar du dem i steg om 32.

Kapitel 3

Dela FileMaker-data med hjälp av JDBC

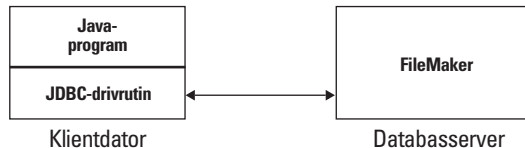
Om du är Java-programmerare kan du använda JDBC-klientdrivrutinen med alla RAD-verktyg (Rapid Application Development) för att visuellt skapa ett Java-program eller Java-appletprogram som ansluter till en FileMaker-datakälla (Windows). Det Java-program eller Java-appletprogram som använder JDBC-klientdrivrutinen har direktåtkomst till informationen i en FileMaker-databasfil.

Om JDBC

JDBC är ett Java-API för SQL-kommandon, standardspråket för åtkomst till relaterade databaser. JDBC är ett namn och inte en akronym, även om man ofta säger att det står för ”Java Database Connectivity” eftersom det är Javas motsvarighet till ODBC. JDBC är ett lågnivågränssnitt vilket betyder att det används för att anropa SQL-kommandon direkt. Det är också utformat för att användas som en bas för gränssnitt och verktyg på hög nivå.

Java-programmet eller Java-appletprogrammet kan kommunicera direkt med en FileMaker-databasfil när du använder JDBC-klientdrivrutinen. Dina SQL-instruktioner sänds till FileMaker-värden för databasfilen och resultatet av instruktionerna sänds tillbaka till dig. Om du använder FileMaker Server som värd, kan den FileMaker-databas som du använder som datakälla ligga på en annan dator (serverdatorn) som är ansluten till nätverket, medan Java-programmet eller klientprogrammet finns på din dator (klientdatorn). Det här kallas för klient-/serverkonfiguration.

Leverantörsspecifika protokoll för DBMS-system



Använda JDBC-klientdrivrutinen

Du kan använda JDBC-klientdrivrutinen med en Java-kompilator eller ett RAD-verktyg för att ansluta till databasen när du skriver koden för Java-programmet eller Java-appletprogrammet. När Java-programmet eller Java-appletprogrammet har skapats måste JDBC-klientdrivrutinen finnas på samma plats som filerna eller inkluderas i koden för att programmet eller appletprogrammet ska kunna kommunicera med databasen.

Om du vill använda JDBC-klienten måste Java-programmet eller Java-appletprogrammet registrera drivrutinen med JDBC-drivrutinshanteraren och du måste ange korrekt JDBC-URL i programmet eller appletprogrammet. Du behöver en JDBC-URL för att kunna ansluta till databasen.

Installera JDBC-klientdrivrutinen

JDBC-klientdrivrutinen levereras som en Java-arkivfil (med filändelsen .jar) och innehåller en samling klassfiler. Arkivfilen heter ddsjlc.jar och finns på installationsskivan, i mappen \xDBC\JDBC Client Driver Installer. I handboken *Installation av FileMaker-drivrutiner för ODBC- och JDBC-klienter* finns information om hur du extraherar innehållet i .jar-filen.

Om JDBC-klientdrivrutinen

JDBC-klientdrivrutinen är en JDBC 3.0 API-kompatibel drivrutin som fungerar med JDK (Java Development Kit) 1.4. Det är en Type 4-drivrutin, dvs. ett ursprungsprotokoll och en äkta Java-drivrutin som konverterar JDBC-anrop direkt till det nätverksprotokoll som FileMaker använder. Denna drivrutinstyp har alla fördelarna med Java inklusive automatisk installation (t.ex. nedladdning av JDBC-drivrutiner med det appletprogram som de används i). Drivrutinen fungerar med JDK 1.3 och Java 2 så länge du endast använder JDBC 1.2-anrop i Java 2-miljö.

Drivrutinsklassen och ingångspunkten för drivrutinen kallas:

```
com.ddtek.jdbc.sequellink.SequeLinkDriver
```

Viktigt! JDBC-klientdrivrutinen ersätter den JDBC-drivrutin för FileMaker som medföljde tidigare versioner av FileMaker. Om du har konfigurerat åtkomst till en FileMaker-datakälla med en äldre drivrutin, måste du definiera om åtkomsten genom att använda och konfigurera den nya drivrutinen.

Använda en JDBC-URL för att ansluta till en databas

I Java har du tillgång till de flesta resurser via webbadresser (URL-adresser). En JDBC-URL används för att identifiera databasen så att JDBC-klientdrivrutinen kan identifiera och upprätta en anslutning till databasen.

En JDBC-URL består av tre huvuddelar som är åtskilda av kolon:

```
jdbc:<delprotokoll>:<delnamn>
```

Första delen i en JDBC-URL är alltid JDBC-protokollet ("jdbc"). *Delprotokollet* är drivrutinsnamnet eller namnet på den mekanism som har stöd för flera drivrutiner. För JDBC-klientdrivrutinen är delprotokollet `sequelink`. *Delnamnet* är IP-adressen till den dator där FileMaker-datakällan finns.

Registrera JDBC-klientdrivrutinen och ansluta till en FileMaker-datakälla (ett exempel)

Här följer ett exempel där JDBCtest:

1. Registrerar JDBC-klientdrivrutinen med JDBC-drivrutinshanteraren
2. Upprättar en anslutning med FileMaker-datakällan (JDBC-URL är `jdbc:sequelink://17.184.17.170:2399`)
3. Returnerar felkoder

```
import java.sql.*;
class FMPJDBCtest
{
    public static void main(String[] args)
    {
        // registrera JDBC-klientdrivrutinen
        try {
            Driver d =
            (Driver)Class.forName("com.ddtek.jdbc.sequellink.SequeLinkDriver").newInstance();
        } catch(Exception e) {
            System.out.println(e);
        }
        // upprätta en anslutning till FileMaker
        Connection con;
        try {
            con =
```

```

DriverManager.getConnection("jdbc:sequelink://17.184.17.170:2399;
user=en användare;password=ett lösenord",userName,password);
} catch(Exception e) {
    system.out.println(e);
}
// hämta anslutningsvarningar
SQLWarning warning = null;
try {
    warning = con.getWarnings();
    if (warning == null) {
        System.out.println("Inga varningar");
        return;
    }
    while (warning != null) {
        System.out.println("Varning: "+warning);
        warning = warning.getNextWarning();
    }
} catch (Exception e) {
    System.out.println(e);
}
}
}
}

```

Obs! Det här exemplet ska inte kompileras.

Specificera drivrutinens egenskaper i URL-delnamnet

Ange drivrutinsegenskaper för användare och lösenord i JDBC-URLens delnamn. Det här är samma egenskaper som kan skickas till anslutningen när metoden `DriverManager.getConnection` anropas via parametern `Egenskaper`.

- användare: ett konto i FileMaker-databasfilen som använder en behörighetsuppsättning med den utökade behörigheten Åtkomst via ODBC/JDBC.
- lösenord: lösenordet för kontot i FileMaker-databasfilen

Grundläggande JDBC URL-anslutning

Utformning: `jdbc:sequelink://<sequelink-värd-IP-adress>:<port>`

Det här är den URL som du använder för att ansluta till FileMaker-databasfilen utan lösenord. Portnumret är alltid 2399 (du kan inte ändra JDBC-delningen till en annan port).

Om du kör JDBC URL-anslutning på samma dator som filen finns på, kan du använda `jdbc:sequelink://localhost:2399`.

Exempel: `jdbc:sequelink://17.184.17.170:2399`

JDBC URL-anslutning med användarnamn och lösenord angivna i URLen

Utformning: `jdbc:sequelink://<sequelink-värd-IP-adress>:<port>;user=användarnamn;password=lösenord`

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

JDBC URL-anslutning med databasnamnet angivet i URLen

Utformning: `jdbc:sequelink://<sequelink-värd-IP-adress>:<port>;databaseName=databasnamn`

Exempel: `jdbc:sequelink://17.184.17.170:2399;databaseName=publications`

JDBC URL-anslutning med databasnamnet, användarnamnet och lösenordet angivet i URLen

Utformning 1 (använd användarnamn och lösenord från datakällan): `jdbc:sequelink://<sequelink-värd-IP-adress>:<port>;databaseName=databasnamn;DBUser=databas användarnamn;DBPassword=databaslösenord`

Utformning 2 (använd användarnamn och lösenord från värddatorn): `jdbc:sequelink://17.184.17.170:2399;databaseName=publications;HUser=användarnamn;HPassword=lösenord`

Obs! SSL-kryptering (Secure Socket Layer) stöds inte. Om du vill skapa en säkrare JDBC-lösning konfigurerar du en miljö som .ASP eller .JSP, där webbservern är värd via https och kommunicerar med FileMaker-databasfilen bakom en brandvägg.

Kontrollera åtkomst via JDBC

När du installerar JDBC-klientdrivrutinen kan du installera JDBCTest som en hjälp att kontrollera anslutningarna. När du installerar JDBC-klientdrivrutinen väljer du Installera utvecklingsverktyg för att installera JDBCTest. I handboken *Installation av FileMaker-drivrutiner för ODBC- och JDBC-klienter* i mappen \xDBC\JDBC Client Driver Installer (på installationsskivan) finns mer information.

När du kontrollerar åtkomsten till en FileMaker-databasfil via JDBC, måste du se till att:

- FileMaker-databasfilen har en värd och är tillgänglig.
- FileMaker-kontot använder en behörighetsuppsättning med den utökade behörigheten Åtkomst via ODBC/JDBC (i FileMaker-databasfilen).
- FileMaker Pro- eller FileMaker Server-värdprogrammet har konfigurerats för delning via ODBC/JDBC.
- JDBC-klientregistreringen och JDBC-URLen är korrekta (drivrutinen kan inkluderas i Java-programmet eller finnas på klientdatorn).

Mer information och exempel där Java och JDBC används för allmänt datautbyte och för publikation av FileMaker-data på webben finns på produktsupportsidorna på webbplatsen för FileMaker, Inc. www.filemaker.com/intl. Du kan använda en genväg till webbplatsen i FileMaker Pro genom att välja **Hjälp > FileMaker på webben**.

Kapitel 4

Standarder som stöds

FileMaker Pro och FileMaker Server använder DataDirect 32-BIT SequeLink 5.4-drivrutinerna när FileMaker-databasfiler används som delade datakällor. När ODBC-klientdrivrutinen används tillsammans med FileMaker Pro eller FileMaker Server stöder den ODBC 3.5 nivå 1 och vissa funktioner i nivå 2. När JDBC-klientdrivrutinen används med FileMaker Pro eller FileMaker Server stöder den JDBC 3.0. ODBC- och JDBC-klientdrivrutinerna stöder grunderna i SQL-92 och vissa högre funktioner i SQL-92. I det här kapitlet beskrivs de SQL-uttryck och -konstruktioner som stöds av ODBC- och JDBC-klientdrivrutinerna när de används tillsammans med FileMaker Pro eller FileMaker Server.

Stöd för Unicode-tecken

ODBC- och JDBC-klientdrivrutinerna stöder Unicode API. Om du däremot skapar ett eget program som använder klientdrivrutinerna, bör du använda Ascii för fältnamn, tabellnamn och filnamn (när något annat än ett Unicode-frågeverktyg eller -program används).

SQL-uttryck och -instruktioner

ODBC- och JDBC-klientdrivrutinerna har stöd för följande SQL-uttryck:

SELECT	DELETE	INSERT	UPDATE
CREATE TABLE	ALTER TABLE	CREATE INDEX	DROP INDEX

Klientdrivrutinerna stöder även FileMaker-datatypsmappning till datatyperna i ODBC SQL och JDBC SQL. I bilaga A, "Mappa FileMaker-fält till ODBC-datatyper" och bilaga B, "Mappa FileMaker-fält till JDBC-datatyper" finns information om datatypskonvertering. Vidare information om hur du skapar SQL-frågor finns i allmänna böcker om SQL.

Obs! ODBC- och JDBC-klientdrivrutinerna känner endast igen den första repetitionen i ett repeterat fält. Drivrutinerna stöder heller inte portaler i FileMaker Pro.

SELECT-uttryck

Använd ett SELECT-uttryck för att ange vilka kolumner du efterfrågar. Slutför SELECT-uttrycket med de kolumnuttryck (samma som fältnamnen) som du vill hämta (t.ex. efternamn). Du kan också skriva en asterisk (*) om du vill hämta alla kolumner. Uttrycken kan innehålla matematiska operationer eller strängmanipulationer (t.ex. LÖN * 1.05).

SELECT-uttrycket kan ha flera olika instruktioner:

```
SELECT [DISTINCT] {* | kolumnuttryck [[AS] kolumnalias],...}
FROM tabellnamn [tabellalias] ...
[ WHERE uttr1 rel_operator uttr2 ]
[ GROUP BY {kolumnuttryck, ...} ]
[ HAVING uttr1 rel_operator uttr2 ]
[ UNION [ALL] (SELECT...) ]
[ ORDER BY {sorteringsuttryck [DESC | ASC]}, ... ]
[ FOR UPDATE [OF {kolumnuttryck, ...}] ]
```

Poster inom hakparenteser är valfria.

Du kan använda kolumnalias för att ge kolumnen ett mer beskrivande namn eller för att förkorta ett långt kolumnnamn. Så här kan du t.ex. tilldela aliaset *avdelning* till kolumnen *avd*:

```
SELECT avd AS avdelning FROM anst
```

Fältnamn kan föregås av tabellnamnet eller tabellalias. Exempel: ANST.EFTERNAMN eller A.EFTERNAMN, där A är alias för tabellen ANST.

Operatören DISTINCT kan föregå det första kolumnuttrycket. Denna operatör eliminerar dubblerade rader från ett frågeresultat. Exempel:

```
SELECT DISTINCT avd FROM anst
```

Använd SQL-instruktionen ...

... om du vill

FROM	Ange vilka tabeller som används i SELECT-uttrycket.
WHERE	Ange de villkor som posterna måste uppfylla för att hämtas (som en FileMaker Pro-sökpost).
GROUP BY	Ange namnen på ett eller flera fält som de returnerade värdena ska grupperas efter. Denna instruktion används för att returnera en uppsättning statistikvärden genom att returnera en rad för varje grupp (som en delsumma i FileMaker Pro).
HAVING	Ange villkor för grupper av poster (t.ex. endast visa de avdelningar som har löner som uppgår till mer än 200 000). Instruktionen är bara giltig om du redan har definierat en GROUP BY-instruktion.
UNION	Kombinera resultatet av två SELECT-uttryck till ett enda resultat. Det odelade resultatet är alla de returnerade posterna från båda SELECT-uttrycken. Som standard returneras inte dubblerade poster. Om du vill returnera dubbla poster använder du nyckelordet ALL (UNION ALL).
ORDER BY	Ange hur posterna är sorterade.
FOR UPDATE	Låsa posterna i databastabellen som valts av SELECT-uttrycket.

Exempel

Följande exempel hämtar namnen på de anställda som tjänar minst 20 000:

```
SELECT efternamn, förnamn FROM anst WHERE lön >= 20000
```

Följande exempel använder instruktionen ORDER BY för att sortera efter både efternamn och förnamn i stigande ordning:

```
SELECT anst_id, efternamn, förnamn FROM anst ORDER BY efternamn, förnamn
```

Fler exempel:

Med	SQL-kod
textkonstant	SELECT 'KattHund' FROM Säljare
numerisk konstant	SELECT 999 FROM Säljare
datumkonstant	SELECT DATE '2004-06-05' FROM Säljare
tidskonstant	SELECT TIME '02:49:03' FROM Säljare
tidsstämpelkonstant	SELECT TIMESTAMP '2004-06-05 02:49:03' FROM Säljare
textkolumn	SELECT Företagsnamn FROM Säljdata SELECT DISTINCT Företagsnamn FROM Säljdata

Med	SQL-kod
numerisk kolumn	<pre>SELECT Belopp FROM Säljdata SELECT DISTINCT Belopp FROM Säljdata</pre>
datumkolumn	<pre>SELECT Försäljningsdatum FROM Säljdata SELECT DISTINCT Försäljningsdatum FROM Säljdata</pre>
tidskolumn	<pre>SELECT Försäljningstid FROM Säljdata SELECT DISTINCT Försäljningstid FROM Säljdata</pre>
tidsstämpelkolumn	<pre>SELECT Tidsstämpel_för_försäljning FROM Säljdata SELECT DISTINCT Tidsstämpel_för_försäljning FROM Säljdata</pre>
BLOB ^a -kolumn	<pre>SELECT Företagslogotyp FROM Säljdata SELECT DISTINCT Företagslogotyp FROM Säljdata</pre>
Jokertecken *	<pre>SELECT * FROM Säljare SELECT DISTINCT * FROM Säljare</pre>

a. En BLOB är ett containerfält i en FileMaker-databasfil.

Information om exemplen

En konstant är ett enskilt, ordagrant värde som du kan ta med i ett SELECT-uttryck medan en kolumn är en referens till ett fält i FileMaker-databasfilen (fältet kan innehålla många olika värden).

Jokertecknet asterisk (*) är ett sätt att ange ”allt”. I exemplet `SELECT * FROM Säljare` är resultatet alla rader i tabellen Säljare. I exemplet `SELECT DISTINCT * FROM Säljare` är resultatet alla unika rader i tabellen Säljare (inga dubletter).

DELETE-uttryck

Använd DELETE-uttryck när du vill ta bort poster från en databastabell. DELETE-uttrycket har följande format:

```
DELETE FROM tabellnamn [ WHERE { villkor } ]
```

WHERE-instruktionen avgör vilka poster som ska raderas. Om du inte inkluderar nyckelordet WHERE raderas alla poster i tabellen (men själva tabellen lämnas intakt).

Ett exempel på ett DELETE-uttryck för tabellen Anställda:

```
DELETE FROM anst WHERE anst_id = 'E10001'
```

DELETE-uttrycket tar bort alla poster som uppfyller villkoren i instruktionen WHERE. I det här fallet raderas alla poster som har anställnings-ID E10001. Eftersom varje anställnings-ID är unikt i tabellen Anställda, raderas endast en post.

INSERT-uttryck

Använd INSERT-uttryck om du vill skapa poster i en databastabell. Du kan ange något av följande:

- En lista över värden som ska infogas som en ny post.
- Ett SELECT-uttryck som kopierar data från en annan tabell som ska infogas som en uppsättning nya poster.

INSERT-uttrycket har följande format:

```
INSERT INTO tabellnamn [(kolumnnamn, ...)] VALUES (uttr, ...)
```

kolumnnamn är en valfri lista över kolumnnamn som ger tillgång till namnet och ordningsföljden för de kolumner vilkas värde anges i instruktionen VALUES. Om du utelämnar kolumnnamn måste värdeuttrycken (uttr) ge värden för alla kolumner som är definierade i tabellen. De måste också komma i samma ordningsföljd som kolumnerna definierades i tabellen.

uttr är den lista över uttryck som ger värdena för den nya postens kolumner. Normalt är uttrycken konstanta värden för kolumnerna (men de kan också vara en delfråga). Värden för teckensträngar måste omslutas av enkla citationstecken ('). Om du vill ta med ett enkelt citationstecken i ett värde för en teckensträng som omsluts av enkla citationstecken, använder du två enkla citationstecken tillsammans (t.ex. 'tro"t'). Datum-, tids- och tidsstämpelvärden måste omslutas av klammerparenteser {}. Logiska värden som är tecken måste omslutas av punkter (t.ex. .S. och .F.). Delfrågor måste omges av parenteser.

I följande exempel infogas en lista över uttryck:

```
INSERT INTO anst (efternamn, förnamn, anst_id, lön, anst_datum)
VALUES ('Andersson', 'Anders', 'E22345', 27500, {2004-06-05})
```

Varje INSERT-uttryck lägger till en post i databastabellen. I det här fallet har en post lagts till i databastabellen över anställda, ANST. Värden har angetts för fem kolumner. De återstående kolumnerna i tabellen tilldelas ett tomt värde, dvs. Null.

SELECT-uttrycket är en fråga som returnerar värden för varje kolumnnamnvärde som anges i listan över kolumnnamn. Att använda ett SELECT-uttryck i stället för en lista över värdeuttryck medför att du kan välja en uppsättning rader från en tabell och infoga den i en annan tabell med ett enda INSERT-uttryck.

Här är ett exempel på ett INSERT-uttryck som använder sig av ett SELECT-uttryck:

```
INSERT INTO anst1 (förnamn, efternamn, anst_id, avd, lön)
SELECT förnamn, efternamn, anst_id, avd, lön FROM anst
WHERE avd = 'D050'
```

I den här typen av INSERT-uttryck måste antalet kolumner som ska infogas matcha antalet kolumner i SELECT-uttrycket. Listan över kolumner som ska infogas måste motsvara kolumnerna i SELECT-uttrycket på samma sätt som den måste motsvara en lista över värdeuttryck i den andra typen av INSERT-uttryck.

Exempel: Den första infogade kolumnen motsvarar den första valda kolumnen; den andra infogade kolumnen motsvarar den andra valda, osv.

Storleken och datatypen för dessa motsvarande kolumner måste överensstämja. Varje kolumn i listan SELECT bör ha en datatyp som ODBC- eller JDBC-drivrutinen accepterar vid en vanlig INSERT/UPDATE av den motsvarande kolumnen i listan INSERT. Värdena trunckeras när storleken på värdet i kolumnen i listan SELECT är större än storleken på den motsvarande kolumnen i listan INSERT.

SELECT-uttrycket beräknas innan några värden infogas.

UPDATE-uttryck

Använd UPDATE-uttryck om du vill ändra poster i en databastabell. UPDATE-uttrycket har följande format:

```
UPDATE tabellnamn SET kolumnnamn = uttr, ... [ WHERE {villkor} ]
```

kolumnnamn är namnet på en kolumn vars värde ska ändras. Det går att ändra flera kolumner i ett och samma uttryck.

uttr är kolumnens nya värde. Normalt är uttrycken konstanta värden för kolumnerna (men de kan också vara en delfråga). Värden för teckensträngar måste omslutas av enkla citationstecken ('). Om du vill ta med ett enkelt citationstecken i ett värde för en teckensträng som omsluts av enkla citationstecken, använder du två enkla citationstecken tillsammans (t.ex. 'tro"t'). Datum-, tids- och tidsstämpelvärden måste omslutas av klammerparenteser {}. Logiska värden som är tecken måste omslutas av punkter (t.ex. .S. och .F.).

Delfrågor måste omges av parenteser.

Instruktionen WHERE är en valfri, giltig instruktion. Den bestämmer vilka poster som uppdateras.

Ett exempel på ett UPDATE-uttryck för tabellen Anställda:

```
UPDATE anst SET lön=32000, undantag=1 WHERE anst_id = 'E10001'
```

Uttrycket UPDATE ändrar alla poster som uppfyller villkoren i instruktionen WHERE. I det här fallet ändras lönen och undantagsstatus för alla anställda som har anställnings-ID E10001. Eftersom varje anställnings-ID är unikt i tabellen Anställda, uppdateras endast en post.

Här följer ett exempel där en delfråga används:

```
UPDATE anst SET lön = (SELECT medel(lön) FROM anst) WHERE anst_id = 'E10001'
```

I det här fallet ändras lönen till medellönen i företaget för de anställda som har anställnings-ID E10001.

CREATE TABLE-uttryck

Använd ett CREATE TABLE-uttryck om du vill skapa en tabell i en databasfil. CREATE TABLE-uttrycket har följande format:

```
CREATE TABLE tabellnamn tabellelementlista [NOT NULL]
```

I instruktionen anger du namnet och datatypen för varje kolumn.

tabellnamn och tabellelementlista har en övre gräns på 100 tecken. Om du anger att en kolumn ska vara NOT NULL aktiveras automatiskt kontrolltillvalet Ej tomt för motsvarande fält i FileMaker-databasfilen. Fältet flaggas med texten Ej tomt på fliken Fält i dialogrutan Definiera databas i FileMaker Pro.

Exempel

Med	SQL-kod
textkolumn	CREATE TABLE T1 (C1 VARCHAR, C2 VARCHAR (50), C3 VARCHAR (1001), C4 VARCHAR (500276))
textkolumn, NOT NULL	CREATE TABLE T1NN (C1 VARCHAR NOT NULL, C2 VARCHAR (50) NOT NULL, C3 VARCHAR (1001) NOT NULL, C4 VARCHAR (500276) NOT NULL)
numerisk kolumn	CREATE TABLE T2 (C1 DECIMAL, C2 DECIMAL (10,0), C3 DECIMAL (7539,2), C4 DECIMAL (497925,301))
datumkolumn	CREATE TABLE T3 (C1 DATE, C2 DATE, C3 DATE, C4 DATE)
tidskolumn	CREATE TABLE T4 (C1 TIME, C2 TIME, C3 TIME, C4 TIME)
tidsstämpelkolumn	CREATE TABLE T5 (C1 TIMESTAMP, C2 TIMESTAMP, C3 TIMESTAMP, C4 TIMESTAMP)
BLOB-kolumn	CREATE TABLE T6 (C1 BLOB, C2 BLOB, C3 BLOB, C4 BLOB)

ALTER TABLE-uttryck

Använd ett ALTER TABLE-uttryck när du vill ändra strukturen i en befintlig tabell i en databasfil. Du kan bara ändra en kolumn i varje instruktion. ALTER TABLE-uttrycket har följande format:

```
ALTER TABLE tabellnamn ADD [COLUMN] kolumndefinition
```

```
ALTER TABLE tabellnamn DROP [COLUMN] kolumnnamn
```

Du måste känna till strukturen i tabellen och hur du vill ändra den innan du använder ALTER TABLE-uttrycket.

Exempel

Om du vill ...	SQL-kod
lägga till kolumner	ALTER TABLE Säljare ADD (C1 VARCHAR)
ta bort kolumner	ALTER TABLE Säljare DROP (C1)

CREATE INDEX-uttryck

Använd ett CREATE INDEX-uttryck när du vill söka snabbare i en databasfil. CREATE INDEX-uttrycket har följande format:

```
CREATE INDEX [ indexnamn ][ON] tabellnamn.kolumnnamn
```

CREATE INDEX kan användas för en enskild kolumn (indexering i flera kolumner stöds inte). Indexeringar kan inte göras på kolumner som motsvarar containerfälttyper, statistikfält, fält som använder tillvalet för global lagring eller beräkningsfält vilkas värden inte lagras i en FileMaker-databasfil.

Om du skapar ett index för en textkolumn aktiveras automatiskt indexeringstillvalet Minimal under Indexering för motsvarande fält i FileMaker-databasfilen. Om du skapar ett index för en icke-textkolumn (eller en kolumn utformad för japansk text) aktiveras automatiskt indexeringstillvalet Allt under Indexering för motsvarande fält i FileMaker-databasfilen.

Om du skapar ett index för en kolumn (vilken som helst) aktiveras automatiskt indexeringstillvalet Skapa index automatiskt vid behov under Indexering för motsvarande fält i FileMaker-databasfilen.

Exempel

```
CREATE INDEX mittIndex ON Säljare.Försäljar_ID
```

DROP INDEX-uttryck

Använd ett DROP INDEX-uttryck när du vill ta bort ett index från en databasfil. DROP INDEX-uttrycket har följande format:

```
DROP INDEX [ON] tabellnamn.kolumnnamn
```

Ta bort ett index när databasfilen är för stor eller om du sällan använder ett fält i frågorna.

Om dina frågor ger dåligt resultat och du arbetar med en mycket stor FileMaker-databasfil med många indexerade textfält, bör du överväga att ta bort index från några av fälten. Du kan också ta bort index från fält som du sällan använder i SELECT-uttryck.

Om du tar bort ett index från en kolumn (vilken som helst) kommer indexeringstillvalet Ingen att aktiveras och rutan Skapa index automatiskt vid behov att avmarkeras under Indexering för motsvarande fält i FileMaker-databasfilen.

Attributet PREVENT INDEX CREATION stöds inte.

Exempel

```
DROP INDEX ON Säljare.Försäljar_ID
```

FROM-instruktion

FROM-instruktionen indikerar att tabellerna ska användas i SELECT-uttrycket. Formatet är:

```
FROM tabellnamn [tabellalias]
```

tabellnamn kan vara ett eller flera enkla tabellnamn i den aktuella arbetsmappen eller fullständiga sökvägar.

Du kan använda tabellalias för att ge tabellen ett mer beskrivande namn eller för att förkorta ett långt tabellnamn. Fältnamn kan föregås av tabellnamnet eller tabellalias. Med specifikationen FROM anställd A kan du t.ex. hänvisa till fältet EFTERNAMN som A.EFTERNAMN. Tabellalias måste användas om SELECT-uttrycket kopplar en tabell till sig själv. Exempel:

```
SELECT * FROM anställd A, anställd F WHERE A.chefs_id = F.anställnings_id
```

Likhetstecknet (=) tar bara med matchande rader i resultatet.

Om du kopplar mer än en tabell och du vill radera alla rader som inte har motsvarande rader i båda källtabellerna kan du använda INNER JOIN. Exempel:

```
SELECT *
FROM Säljare INNER JOIN Säljdata
ON Säljare.Försäljar_ID = Säljdata.Försäljar_ID
```

Om du kopplar mer än en tabell och du vill inkludera ej matchande rader från den första tabell du namnger, kan du använda LEFT OUTER JOIN. Exempel:

```
SELECT *
FROM Säljare LEFT OUTER JOIN Säljdata
ON Säljare.Försäljar_ID = Säljdata.Försäljar_ID
```

WHERE-instruktion

WHERE-instruktionen anger vilka villkor som poster måste uppfylla för att kunna hämtas. WHERE-instruktionen innehåller villkor i formatet:

```
WHERE uttr1 rel_operator uttr2
```

uttr1 och uttr2 kan vara fältnamn, konstantvärden eller uttryck.

rel_operator är den relationsoperator som kopplar ihop de två uttrycken. Följande SELECT-uttryck hämtar t.ex. namnen på de anställda som tjänar minst 200 000 kronor.

```
SELECT efternamn, förnamn FROM anst WHERE lön >= 200000
```

GROUP BY-instruktion

GROUP BY-instruktionen anger namnet på ett eller flera fält som de returnerade värdena ska grupperas efter. Du använder denna instruktion för att returnera en uppsättning statistikvärden. Instruktionen har följande format:

```
GROUP BY kolumnuttryck
```

kolumnuttryck måste matcha det kolumnuttryck som används i SELECT-uttrycket. Ett kolumnuttryck kan vara antingen ett eller flera fältnamn i databastabellen avgränsade med kommatecken eller ett eller flera uttryck avgränsade med kommatecken.

Följande exempel summerar lönerna på varje avdelning.

```
SELECT avd_id, SUM(lön) FROM anst GROUP BY avd_id
```

Denna instruktion returnerar en rad för varje specifikt avdelnings-ID. Varje rad innehåller avdelnings-ID och summan av lönerna för de anställda på avdelningen.

HAVING-instruktion

Med HAVING-instruktionen kan du ange villkor för grupper av poster (t.ex. endast visa de avdelningar som har löner som uppgår till mer än 200 000). Instruktionen är bara giltig om du redan har definierat en GROUP BY-instruktion. Instruktionen har följande format:

```
HAVING uttr1 rel_operator uttr2
```

uttr1 och uttr2 kan vara fältnamn, konstantvärden eller uttryck. Dessa uttryck måste inte matcha ett kolumnuttryck i SELECT-instruktionen.

rel_operator är den relationsoperator som kopplar ihop de två uttrycken. Följande exempel returnerar endast de avdelningar vilkas lönesummor är större än 200 000:

```
SELECT avd_id, SUM(lön) FROM anst
GROUP BY avd_id HAVING SUM(lön) > 200000
```

UNION-operator

UNION-operatorn kombinerar resultatet av två SELECT-uttryck till ett enda resultat. Det odelade resultatet är alla de returnerade posterna från båda SELECT-uttrycken. Som standard returneras inte dubblerade poster. Om du vill returnera dubblerade poster använder du nyckelordet ALL (UNION ALL). Formatet är:

```
SELECT-uttryck UNION [ALL] SELECT-uttryck
```

När du använder UNION-operatorn måste urvalslistorna för varje SELECT-uttryck ha samma antal kolumnuttryck, med samma datatyper och anges i samma ordningsföljd. Exempel:

```
SELECT efternamn, lön, anst_datum FROM anst UNION SELECT namn, lön, födelsedatum
FROM person
```

Detta exempel har samma antal kolumnuttryck och varje kolumnuttryck, i ordningsföljd, har samma datatyp. Följande exempel är inte giltigt eftersom datatyperna för kolumnuttrycken är olika (LÖN från ANST har en annan datatyp än EFTERNAMN från LÖNEFÖRHÖJNING). Detta exempel har samma antal kolumnuttryck i varje SELECT-uttryck, men uttrycken är inte i samma ordningsföljd som datatypen.

```
SELECT efternamn, lön FROM anst UNION SELECT lön, efternamn FROM löneförhöjning
```

ORDER BY-instruktion

ORDER BY-instruktionen anger hur posterna ska sorteras. Formatet är:

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

sorteringsuttryck kan vara fältnamn, uttryck eller positionsnummer för det kolumnuttryck som ska användas. Som standard sker sorteringen i stigande ordning (ASC).

Om du t.ex. vill sortera efter efternamn och sedan efter förnamn använder du ett av följande SELECT-uttryck:

```
SELECT anst_id, efternamn, förnamn FROM anst ORDER BY efternamn, förnamn
eller
```

```
SELECT anst_id, efternamn, förnamn FROM anst ORDER BY 2,3
```

I det andra exemplet är efternamn det andra kolumnuttrycket som följer efter SELECT, så ORDER BY 2 sorterar efter efternamn.

FOR UPDATE-instruktion

FOR UPDATE-instruktionen läser posterna i databastabellen som valts av SELECT-uttrycket. Formatet är:

```
FOR UPDATE [OF kolumnuttryck]
```

kolumnuttryck är en lista över de fältnamn i databastabellen som du vill uppdatera, avgränsade av kommatecken. kolumnuttryck är valfritt.

Följande exempel returnerar alla poster i databasen över anställda som har ett värde i fältet LÖN som är högre än 20 000. När varje post hämtas är den låst. Om posten uppdateras eller raderas är posten låst tills du har gjort ändringen. I annat fall läses den upp när du hämtar nästa post.

```
SELECT * FROM anst WHERE lön > 20000 FOR UPDATE OF efternamn, förnamn, lön
```

Statistikfunktioner i SQL

Statistikfunktioner returnerar ett enskilt värde från en uppsättning poster. Du kan använda en statistikfunktion som en del av ett SELECT-uttryck, med ett fältnamn (t.ex. AVG(lön) eller i kombination med ett kolumnuttryck (t.ex. AVG(lön * 1.07)).

Du kan låta kolumnuttrycket föregås av DISTINCT-operatoren för att eliminera dubblerade värden. Exempel:

```
COUNT (DISTINCT efternamn)
```

I det här exemplet räknas endast unika värden som efternamn.

Statistikfunktionen returnerar följande

SUM	Totalsumman för värdena i det numeriska fältuttrycket. Till exempel returnerar SUM(Lön) summan av alla värden i fältet Lön.
AVG	Medelvärde för värdena i ett numeriskt fältuttryck. Till exempel returnerar AVG(Lön) medelvärdet av alla värden i fältet Lön.
COUNT	Antal värden i ett fältuttryck. Till exempel returnerar COUNT(Namn) antalet värden i fältet Namn. När du använder COUNT med ett fältnamn returnerar COUNT antalet fältvärden som inte är tomma. Ett specialexempel är COUNT(*), som returnerar antal poster i uppsättningen, även tomma poster.
MAX	Maxvärdet för ett fältuttryck. Till exempel returnerar MAX(Lön) det maximala värdet i fältet Lön.
MIN	Minimivärdet för ett fältuttryck. Till exempel returnerar MIN(Lön) det minsta värdet i fältet Lön.

Exempel

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

SQL-uttryck

Använd uttryck i WHERE-, HAVING- och ORDER BY-instruktioner i SELECT-uttryck när du vill skapa detaljerade och avancerade databasfrågor.

Följande giltiga uttryckselement finns:

- Fältnamn
- Konstanter och ordagranna strängar
- Exponentnotation
- Numeriska operatorer
- Teckenoperatorer
- Datumoperatorer
- Relationsoperatorer
- Logiska operatorer
- Funktioner

Fältnamn

Det vanligaste uttrycket är ett enkelt fältnamn, t.ex. `beräkn` eller `Säljdata.Faktura_ID`.

Konstanter och ordagranna strängar

Konstanter är värden som inte ändras. I uttrycket `PRIS * 1.05` är värdet `1.05` en konstant. Du kan också använda värdet `30` i konstanten `Antal_dagar_i_juni`.

En ordagrann sträng är en annan typ av konstant, men i stället för att ha ett tilldelat värde är själva strängen värdet, t.ex. `'Paris'` eller `'14:35:10'`. En ordagrann sträng är en typ av ”vad du ser är vad du får”-konstant.

Teckenkonstanter (t.ex. ordagranna strängar) måste omslutas av enkla citationstecken (`'`). Om du vill ta med ett enkelt citationstecken i en teckenkonstant som omsluts av enkla citationstecken, använder du två enkla citationstecken tillsammans (t.ex. `'tro''t'`).

Du måste omsluta datum-, tid- och tidsstämpelkonstanter med klammerparenteser (`{ }`), t.ex. `{D '2005-06-05'}`, `{14:35:10}` och `{TS '2005-06-05 14:35:10'}`. Det enda undantaget: För SQL-92-syntaxen ska ISO-datum och ISO-klockslag vara utan parenteser:

- `DATE 'YYYY-MM-DD'`
- `TIME 'HH:MM:SS'`
- `TIMESTAMP 'YYYY-MM-DD HH:MM:SS'`

Konstant	Giltig syntax (exempel)
Text	Paris'
Tal	1.05
Datum	DATE '2005-06-05' { D '2005-06-05' } { 06/05/2005 } { 06/05/05 }
Klockslag	TIME '14:35:10' { T '14:35:10' } { 14:35:10 }
Tidsstämpel	TIMESTAMP '2005-06-05 14:35:10' { TS '2005-06-05 14:35:10' } { 06/05/2005 14:35:10 } { 06/05/05 14:35:10 } { 2005-06-05 14:35:10 }

Logiska värden som utgörs av tecken måste omslutas av punkter. De logiska konstanterna är .T. och 1 för sant och .F. och 0 för falskt. För enkelhetens skull bör du använda 1 och 0.

Exponentiell/matematisk notation

Du kan använda exponentiell notation.

Exempel

```
SELECT kolumn1, 3.4E+7 FROM tabell1 WHERE beräkn < 3.4E-6 * kolumn2
```

Numeriska operatörer

Du kan ta med följande operatörer i numeriska uttryck: +, -, *, /, ^ och ** (exponent).

Du kan låta numeriska uttryck föregås av ett unärt plus (+) eller minus (-).

Teckenoperatörer

Du kan sammanlänka tecken.

Exempel

I följande exempel är efternamn 'JANSSON ' och förnamn 'ROBERT ':

Operator	Sammanlänkning	Exempel	Resultat
+	Behåll avslutande tomma tecken	förnamn + efternamn	'ROBERT JANSSON '
-	Flytta avslutande tomma tecken till slutet	förnamn - efternamn	'ROBERTJANSSON '

Datumoperatorer

Du kan ändra datum.

Exempel

I exemplen nedan är anst_datum {01/30/2004}

Operator	Effekt på datum	Exempel	Resultat
+	Lägg till ett antal dagar i ett datum	anst_datum + 5	{02/04/2004}
-	Ta reda på antalet dagar mellan två datum eller subtrahera antalet dagar från ett datum	anst_datum - {01/01/2004} anst_datum - 10	29 {01/20/2004}

Fler exempel:

```
SELECT Försäljningsdatum, Försäljningsdatum + 30 AS agg FROM Säljdata
SELECT Försäljningsdatum, Försäljningsdatum - 30 AS agg FROM Säljdata
```

Relationsoperatorer

Operator	Betydelse
=	Lika med
<>	Inte lika med
>	Större än
>=	Större än eller lika med
<	Mindre än
<=	Mindre än eller lika med
LIKE	Matcha ett mönster
NOT LIKE	Matcha inte ett mönster
IS NULL	Lika med Null
IS NOT NULL	Inte lika med Null
BETWEEN	Intervall av värden mellan en undre och en övre gräns
IN	En medlem av en uppsättning angivna värden eller en medlem av en delfråga
NOT IN	Inte en medlem av en uppsättning angivna värden eller en medlem av en delfråga
EXISTS	”Sant” om en delfråga returneras som minst en post
ANY	Jämför ett värde med varje värde som returneras av en delfråga (operatorm måste föregås av =, <>, >, >=, < eller <=); =ANY motsvarar IN
ALL	Jämför ett värde med varje värde som returneras av en delfråga (operatorm måste föregås av =, <>, >, >=, < eller <=)

Exempel

```

SELECT Säljdata.Faktura_ID FROM Säljdata WHERE Säljdata.Försäljar_ID = 'FS-1'
SELECT Säljdata.Summa FROM Säljdata WHERE Säljdata.Faktura_ID <> 125
SELECT Säljdata.Summa FROM Säljdata WHERE Säljdata.Summa > 3000
SELECT Säljdata.Försäljningstid FROM Säljdata WHERE Säljdata.Försäljningstid
  < '12:00:00'
SELECT * FROM Säljdata WHERE Säljdata.Företagsnamn LIKE '%Universitet'
SELECT * FROM Säljdata WHERE Säljdata.Företagsnamn NOT LIKE '%Universitet'
SELECT * FROM Säljdata WHERE Säljdata.Summa IS NULL
SELECT * FROM Säljdata WHERE Säljdata.Summa IS NOT NULL
SELECT * FROM Säljdata WHERE Säljdata.Faktura_ID BETWEEN 1 AND 10
SELECT COUNT(Säljdata.Faktura_ID) AS agg
  FROM Säljdata WHERE Säljdata.Faktura_ID IN (50,250,100)
SELECT COUNT(Säljdata.Faktura_ID) AS agg
  FROM Säljdata WHERE Säljdata.Faktura_ID NOT IN (50,250,100)
SELECT COUNT (Säljdata.Faktura_ID) AS agg FROM Säljdata
  WHERE Säljdata.Faktura_ID NOT IN (SELECT Säljdata.Faktura_ID
  FROM Säljdata WHERE Säljdata.Försäljar_ID = 'FS-4')
SELECT *
  FROM Säljdata WHERE EXISTS (SELECT Säljdata.Summa
  FROM Säljdata WHERE Säljdata.Försäljar_ID IS NOT NULL)
SELECT *
  FROM Säljdata WHERE Säljdata.Summa = ANY (SELECT Säljdata.Summa
  FROM Säljdata WHERE Säljdata.Försäljar_ID = 'FS-1')
SELECT *
  FROM Säljdata WHERE Säljdata.Summa = ALL (SELECT Säljdata.Summa
  FROM Säljdata WHERE Säljdata.Försäljar_ID IS NULL)

```

Logiska operatörer

Du kan kombinera två eller flera villkor. Relationer måste skapas mellan villkoren med hjälp av AND eller OR, t.ex.:

```
lön = 40000 AND undantag = 1
```

Du använder den logiska NOT-operatören för att göra innebörden till den omvända, t.ex.:

```
NOT (lön = 40000 AND undantag = 1)
```

Exempel

```

SELECT *
  FROM Säljdata
  WHERE Säljdata.Företagsnamn NOT LIKE '%Universitet' AND Säljdata.Summa > 3000
SELECT *
  FROM Säljdata
  WHERE (Säljdata.Företagsnamn LIKE '%Universitet' OR Säljdata.Summa > 3000)
  AND Säljdata.Försäljar_ID = 'FS-1'

```

Funktioner

ODBC- och JDBC-klientdrivrutinerna stöder många funktioner som du kan använda i uttryck. Vissa av funktionerna returnerar teckensträngar, vissa returnerar siffror och vissa returnerar datum.

Funktioner som returnerar teckensträngar	Beskrivning	Exempel
CHR	Konverterar en Ascii-kod till en sträng bestående av ett tecken	CHR(67) returnerar C
RTRIM	Tar bort avslutande blanksteg från en sträng	RTRIM('ABC ') returnerar ABC
TRIM	Tar bort avslutande blanksteg från en sträng	TRIM('ABC ') returnerar ABC
LTRIM	Tar bort inledande blanksteg från en sträng	LTRIM(' ABC') returnerar ABC
UPPER	Ändrar varje bokstav i en sträng till versal	UPPER('Allan') returnerar ALLAN
LOWER	Ändrar varje bokstav i en sträng till gemen	LOWER('Allan') returnerar allan
LEFT	Returnerar tecknen längst till vänster i en sträng	LEFT('Mattsson',3) returnerar Mat
RIGHT	Returnerar tecknen längst till höger i en sträng	RIGHT('Mattsson',4) returnerar sson
SUBSTR	Returnerar en delsträng i en sträng, med parametrar för strängen, det första tecknet att extrahera och antal tecken som ska extraheras (valfritt)	SUBSTR('Konrad',2,3) returnerar onr SUBSTR('Konrad',2) returnerar onrad
SPACE	Genererar en sträng med blanksteg	SPACE(5) returnerar ' '
STRVAL	Konverterar ett värde av vilken typ som helst till en teckensträng	STRVAL('Hallman') returnerar Hallman STRVAL(5 * 3) returnerar 15 STRVAL(4 = 5) returnerar 'Falskt' STRVAL({12/25/2004}) returnerar 12/25/2004
TIME	Returnerar klockslaget som en sträng	Klockan 21:49 returnerar TIME() 21:49:00
USERNAME	Returnerar det inloggnings-ID som angavs vid anslutningstiden	

Exempel

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

```

SELECT SUBSTR(Säljare.Försäljar_ID, 2) +
       SUBSTR(Säljare.Försäljar_ID, 4) AS agg FROM Säljare
SELECT SPACE(2) + Säljare.Försäljar_ID AS Försäljar_ID FROM Säljare
SELECT STRVAL('60506') AS agg FROM Säljdata WHERE Säljdata.Faktura_ID = 1
SELECT TIME() AS agg FROM Säljdata WHERE Säljdata.Faktura_ID = 1
SELECT USERNAME() AS agg FROM Säljdata WHERE Säljdata.Faktura_ID = 1

```

Funktioner som returnerar siffror	Beskrivning	Exempel
MOD	Dividerar två tal och returnerar återstoden av divisionen	MOD(10,3) returnerar 1
LEN	Returnerar längden på en sträng	LEN('ABC') returnerar 3
MONTH	Returnerar månaden i ett datum	MONTH({01/30/2004}) returnerar 1
DAY	Returnerar dagen i ett datum	DAY({01/30/2004}) returnerar 30
YEAR	Returnerar året i ett datum	YEAR({01/30/2004}) returnerar 2004
MAX	Returnerar det högsta av två tal	MAX(66,89) returnerar 89
DAYOFWEEK	Returnerar veckodagen (1-7) i ett datumuttryck	DAYOFWEEK({05/01/2004}) returnerar 7
MIN	Returnerar det lägsta av två tal	MIN(66,89) returnerar 66
POW	Upphöjer ett tal till angiven exponent	POW(7,2) returnerar 49
INT	Returnerar heltalsdelen av ett tal	INT(6.4321) returnerar 6
X	Returnerar decimalmotsvarigheten i ett hexadecimalt tal	X'b9' returnerar 185
B	Returnerar decimalmotsvarigheten i ett binärt tal	B'1001' returnerar 9
ROUND	Avrundar ett tal	ROUND(123.456,0) returnerar 123 ROUND(123.456,2) returnerar 123,46 ROUND(123.456,-2) returnerar 100
NUMVAL	Konverterar en teckensträng till ett tal. Om teckensträngen inte är ett giltigt tal returneras 0.	NUMVAL('123') returnerar 123
VAL	Konverterar en teckensträng till ett tal. Om teckensträngen inte är ett giltigt tal returneras 0.	VAL('123') returnerar 123

Funktioner som returnerar datum	Beskrivning	Exempel
DATE	Returnerar dagens datum	Om dagens datum är 11/11/2004, returnerar DATE() {11/11/2004}
DATEVAL	Konverterar en teckensträng till ett datum	DATEVAL('01/30/2004') returnerar {30/01/2004}

Prioritetsordning för operatörer

Ju mer komplexa uttrycken blir, desto viktigare är den ordning i vilken de utvärderas. I den här tabellen visas ordningsföljden för de operatörer som utvärderas. Operatörerna på den första raden utvärderas först, sedan operatörerna på den andra raden, osv. Operatörer på samma rad utvärderas från vänster till höger i uttrycket.

Prioritetsordning Operator

1	Unärt '-', Unärt '+'
2	^, **
3	*, /
4	+, -
5	=, <>, <, <=, >, >=, LIKE, NOT LIKE, IS NULL, IS NOT NULL, BETWEEN, IN, EXISTS, ANY, ALL
6	NOT
7	AND
8	OR

I följande exempel visas vikten av prioritetsordning:

```
WHERE lön > 40000 OR anst_datum > {01/30/2004} AND avd = 'D101'
```

Eftersom AND utvärderas först hämtar denna fråga alla anställda på avdelning D101 som anställdes efter 30 januari 2004 samt alla anställda som tjänar mer än 40 000, oavsett avdelning och anställningsdatum.

Om du vill tvinga instruktionen att utvärderas i en annan ordning, omsluter du de villkor du vill utvärdera först inom parentes. Exempel:

```
WHERE (lön > 40000 OR anst_datum > {01/30/2004}) AND avd = 'D101'
```

hämtar anställda på avdelning D101 som antingen tjänar mer än 40 000 eller som anställdes efter 30 januari 2004.

ODBC-katalogfunktioner

ODBC-klientdrivrutinen stöder följande katalogfunktioner:

- SQLGetTypeInfo
- SQLTables
- SQLTablePrivileges
- SQLColumns
- SQLColumnPrivileges
- SQLStatistics – rapporterar kolumner med index och kolumner som har markerats som unika, som unika index. Det finns inga fallande index.
- SQLSpecialColumns – RowId rapporteras som NUMBER(20) - ULONGLONG, värdet ändras aldrig och återanvänds aldrig. Icke verkställda infogningar har ett RowId.

Kataloginformation lagras och rapporteras som enstaka namn (endast tabellnamn).

Metadatafunktioner för JDBC

JDBC-klientdrivrutinen stöder följande metadatafunktioner:

- `getTypeInfo`
- `getTables`
- `getTableTypes`
- `getTablePrivileges`
- `getVersionColumns`
- `getBestRowIdentifier`
- `getColumns`

För JDBC ställer du metadatafrågor med kommandot `dmd.getFunction` i Java.

Bilaga A

Mappa FileMaker-fält till ODBC-datatyper

Tabellen nedan illustrerar hur FileMaker-fält mappas till vanliga ODBC-datatyper.

FileMaker-datatype	Konverteras till ODBC-datatype	Om datatypen
text	SQL_VARCHAR	Den maximala kolumnlängden för text är 1 miljon tecken, såvida du inte anger ett värde för Maximalt antal tecken för textfältet i FileMaker. FileMaker returnerar tomma strängar som NULL.
nummer	SQL_DECIMAL	
datum	SQL_DATE	
tid	SQL_TIME	FileMaker-datatypen för tidsfält kan innehålla ett klockslag eller ett tidsintervall. Ett tidsintervall returneras som ett klockslag, såvida det inte är mindre än 0 eller mer än 24 timmar (båda returnerar värdet 0).
tidsstämpel	SQL_TIMESTAMP	Utformningen för tidsstämpel i FileMaker är kompatibel med ODBC SQL-utformningen: [åååå.mm.dd hh:mm:ss]
container (BLOB)	SQL_LONGVARBINARY	Containerfält i FileMaker kan innehålla flera olika typer av data. En post kan innehålla en bild och en annan ett OLE-objekt (Windows). När du delar en databasfil som en datakälla via ODBC konverteras innehållet i containerfält till Jpeg-data.
beräkning		Resultatet mappas till motsvarande ODBC-datatype.

Stränglängden är valfri i tabelluttryck. Alla strängar lagras och hämtas i Unicode.

Obs!

- Du kan hämta upp till 1 024 fält på en och samma gång från en FileMaker-databasfil.
- FileMaker har stöd för repeterade fält (data i tabellformat) medan ODBC saknar detta stöd. FileMaker exporterar repetitioner till tabb- eller kommaavgränsade filer och avgränsar varje repetition med en gruppavgränsare (Unicode-decimalvärdet 29). Textkolumner som avgränsas med gruppavgränsaren länkas samman. Alla andra datatyper returnerar endast den första repetitionen.
- För numeriska datatyper och datatyperna datum, tid, tidsstämpel och container returneras 0.

Bilaga B

Mappa FileMaker-fält till JDBC-datatyper

JDBC-klientdrivrutinen använder följande mappningar vid konvertering av FileMaker-datatyper till JDBC SQL-typer. (Mer information om dessa typer finns på dokumentationssidorna om JDK 1.4 på www.javasoft.com.)

FileMaker-datatype	Konverteras till JDBC SQL-typ
text	java.sql.Types.VARCHAR
nummer	java.sql.Types.DECIMAL
datum	java.sql.Types.DATE
tid	java.sql.Types.TIME
tidsstämpel	java.sql.Types.TIMESTAMP
container	java.sql.Types.BLOB
beräkning	anges av beräkningsresultatets datatype

JDBC-klientdrivrutinen konverterar FileMaker-datatype *beräkning* till den JDBC SQL-typ som matchar beräkningens resultat. JDBC-klientdrivrutinen konverterar t.ex. en FileMaker-beräkning som resulterar i datatype *tidsstämpel* till `java.sql.Types.TIMESTAMP`.

Bilaga C

ODBC- och JDBC-felmeddelanden

Här visas grundutformningarna för de felmeddelanden som kan komma att visas när du arbetar med FileMaker och ODBC/JDBC. En lista över felnummer och förklaringar finns på webbplatsen www.datadirect-technologies.com.

Mer information om hur du löser fel i FileMaker finns där funktionerna Get(SenasteFel) och Get(SenasteODBCfel) beskrivs i FileMaker Pro Hjälp.

ODBC-felmeddelanden

Felmeddelanden kan härröra från:

- ODBC-drivrutinen
- ODBC-drivrutinshanteraren
- SequeLink-klienten
- SequeLink-servern
- Datakällan eller databashanteringssystemet

Felmeddelanden som rör ODBC-drivrutinen

Ett fel som rapporteras av SequeLink ODBC-drivrutinen har följande utformning:

[DataDirect] [ODBC SequeLink-drivrutin] meddelande

Exempel:

[DataDirect] [ODBC SequeLink-drivrutin] Ogiltig precision

Om du får den här typen av felmeddelande bör du kontrollera det senaste ODBC-anropet programmet gjorde och se om du kan upptäcka felet, eller kontakta tillverkaren för ODBC-programmet.

Felmeddelanden som rör ODBC-drivrutinshanteraren

Ett fel som rapporteras av ODBC-drivrutinshanteraren har följande utformning:

[Microsoft] [ODBC-drivrutinshanterare] meddelande

Exempel:

[Microsoft] [ODBC-drivrutinshanterare] Funktionssekvensfel

Om denna typ av fel uppstår bör du kontrollera att du har rätt ODBC-filer och -drivrutiner.

Felmeddelanden som rör SequeLink-klienten

Ett fel som rapporteras av SequeLink ODBC-klienten har följande utformning:

[DataDirect] [ODBC SequeLink-drivrutin] [SequeLink-klient] meddelande

Exempel:

[DataDirect] [ODBC SequeLink-drivrutin] [SequeLink-klient] Det går inte att hitta den angivna omskrivningsmodulen

Felmeddelanden som rör SequeLink-servern

Ett fel som rapporteras av SequeLink-servern har följande utformning:

[DataDirect] [ODBC SequeLink-drivrutin] [SequeLink-server] meddelande

Exempel:

[DataDirect] [ODBC SequeLink-drivrutin] [SequeLink-server] Endast SELECT-uttryck är tillåtna i den skrivskyddade anslutningen.

Felmeddelanden som rör datakällan

Ett felmeddelande som inträffar i datakällan omfattar datakällans namn i följande utformning:

[DataDirect] [ODBC SequeLink-drivrutin] [datakälla] meddelande

Du kan t.ex. få följande meddelande från en FileMaker-datakälla:

[DataDirect] [ODBC SequeLink-drivrutin] [FileMaker] Ogiltigt användarnamn/lösenord

Om du får den här typen av felmeddelande har du gjort något felaktigt med databassystemet. Mer information finns i FileMaker-dokumentationen. Du kan också kontakta din databasadministratör.

JDBC-felmeddelanden

SequeLink för JDBC-drivrutinen rapporterar fel till det anropande programmet genom att returnera SQLExceptions. Felmeddelanden kan härröra från:

- JDBC-drivrutinen
- SequeLink-servern
- Datakällan eller databashanteringssystemet

Felmeddelanden som rör JDBC-drivrutinen

Ett fel som rapporteras av JDBC-drivrutinen har följande utformning:

[DataDirect] [SequeLink JDBC-drivrutin] meddelande

Exempel:

[DataDirect] [SequeLink JDBC-drivrutin] Tidsgränsen har överskridits

Om du får den här typen av felmeddelande bör du kontrollera det senaste JDBC-anropet programmet gjorde och se om du kan upptäcka felet, eller kontakta tillverkaren för ODBC-programmet.

Felmeddelanden som rör SequeLink-servern

Ett fel som rapporteras av SequeLink-servern har följande utformning:

[DataDirect] [SequeLink JDBC-drivrutin] [SequeLink] meddelande

Om inga SequeLink-serverfel har uppstått visas:

[DataDirect] [JDBC SequeLink-drivrutin] [SequeLink]

Felmeddelanden som rör datakällan

Ett felmeddelande som inträffar i datakällan omfattar datakällans namn i följande utformning:

[DataDirect] [SequeLink JDBC-drivrutin] [datakälla] meddelande

Du kan t.ex. få följande meddelande från en FileMaker-datakälla:

[DataDirect] [SequeLink JDBC-drivrutin] [FileMaker] Ogiltigt användarnamn/lösenord

Om du får den här typen av felmeddelande har du gjort något felaktigt med databassystemet. Mer information finns i FileMaker-dokumentationen. Du kan också kontakta din databasadministratör.

Index

A

ALTER TABLE (SQL-uttryck) 22

C

CREATE INDEX (SQL-uttryck) 22

CREATE TABLE (SQL-uttryck) 21

D

datakälla

- använda en FileMaker-databasfil 6
- ett DSN-namn för varje FileMaker-databasfil 7
- inaktivera en delad FileMaker-databasfil 6
- konfigurera för åtkomst via JDBC 15
- konfigurera för åtkomst via ODBC 10
- kontrollera åtkomst via JDBC 16
- kontrollera åtkomst via ODBC 11

datatypsmappning

- JDBC-klientdrivrutin 37
- ODBC-klientdrivrutin 35

datumoperatörer i SQL-uttryck 28

dela en FileMaker-datakälla 6

DELETE (SQL-uttryck) 19

drivrutinsegenskaper

- JDBC-klientdrivrutin 15
- ODBC-klientdrivrutin 10

DROP INDEX (SQL-uttryck) 22

E

exponentiell notation i SQL-uttryck 27

F

FOR UPDATE (SQL-instruktion) 25

FROM (SQL-instruktion) 23

funktioner i SQL-uttryck 30

fältnamn i SQL-uttryck 26

G

GROUP BY (SQL-instruktion) 23

H

HAVING (SQL-instruktion) 24

I

inaktivera en delad FileMaker-databasfil 6

INSERT (SQL-uttryck) 19

J

JDBC, beskrivning 13

JDBC-klientdrivrutin

- ange JDBC-URL 14
- drivrutinsklass och ingångspunkt 14
- installera 13

kontrollera åtkomst 16

mappa datatyper 37

metadatafunktioner 33

registrera med JDBC-drivrutinshanteraren 14

repeterade fält 17

Unicode-stöd 17

JDK (Java Development Kit) 14

K

katalogfunktioner för ODBC 32

klientdrivrutiner som ingår 7

klientprogram, använda FileMaker 6

konfigurera en FileMaker-datakälla

via JDBC 15

via ODBC 10

konstanter i SQL-uttryck 26

kontrollera åtkomst

JDBC-klientdrivrutin 16

ODBC-klientdrivrutin 11

krav för nätverk 6

L

LDAC (Local Data Access Companion) 7

logiska operatörer i SQL-uttryck 29

M

mappa datatyper

JDBC-klientdrivrutin 37

ODBC-klientdrivrutin 35

matematisk notation i SQL-uttryck 27

metadatafunktioner för JDBC 33

Microsoft Access, klientprogram 12

N

Namn på datakälla 11

numeriska operatörer i SQL-uttryck 27

O

ODBC

beskrivning 9

repeterade fält 35

ODBC-klientdrivrutin

installera 10

katalogfunktioner 32

kontrollera åtkomst 11

mappa datatyper 35

maximalt antal FileMaker-fält 35

repeterade fält 17

Unicode-stöd 17

operativsystem 6

ordagranna strängar i SQL-uttryck 26

ORDER BY (SQL-instruktion) 24

P

prioritetsordning för operatörer i SQL-uttryck 32

R

RAD-verktyg (Rapid Application Development) 13

RDAC (Remote Data Access Companion) 7

registrera JDBC-klientdrivrutinen 14

relationsoperatörer i SQL-uttryck 28

repeterade fält 17, 35

S

SELECT (SQL-uttryck) 17

SequeLink-serverport 11

SequeLink-servervärd 11

Serverdatakälla 11

SQL-92-stöd 17

SQL-statistikfunktioner 25

SQL-uttryck 26

ALTER TABLE 22

CREATE INDEX 22

CREATE TABLE 21

datumoperatörer 28

DELETE 19

DROP INDEX 22

exponentiell och matematisk notation 27

funktioner 30

fältnamn 26

INSERT 19

konstanter 26

logiska operatörer 29

numeriska operatörer 27

ordagranna strängar 26

prioritetsordning för operatörer 32

relationsoperatörer 28

SELECT 17

som stöds av klientdrivrutiner 17

teckenoperatörer 27

UPDATE 20

SSL-kryptering 16

statistikfunktioner i SQL 25

T

teckenoperatörer i SQL-uttryck 27

testa åtkomst

JDBC-klientdrivrutin 16

ODBC-klientdrivrutin 11

U

Unicode-stöd 17

UNION (SQL-operator) 24

UPDATE (SQL-uttryck) 20

URL-adress (Uniform Resource Locator)

för JDBC-klientdrivrutinen 14

uttryck i SQL 26

V

värd för en FileMaker-datakälla 6

W

WHERE (SQL-instruktion) 23