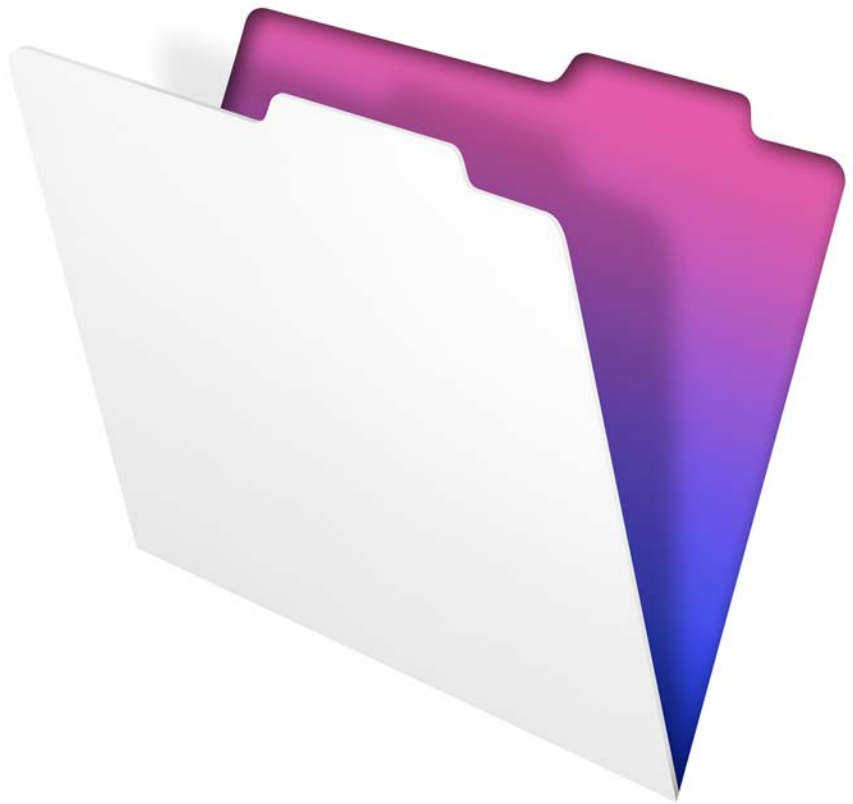


# FileMaker® 11

## Guida ODBC e JDBC



© 2004–2010 FileMaker, Inc. Tutti i diritti riservati.

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

FileMaker è un marchio di FileMaker, Inc. registrato negli Stati Uniti e in altri Paesi. Il logo della cartella dei file è un marchio di FileMaker, Inc. Tutti gli altri marchi appartengono ai rispettivi proprietari.

La documentazione di FileMaker è protetta da copyright. Non è permesso fare copie o distribuire questa documentazione senza previa autorizzazione scritta di FileMaker. È possibile utilizzare la presente documentazione soltanto unitamente a una copia del software FileMaker concessa in licenza.

Tutte le persone, le società, gli indirizzi e-mail e gli URL elencati negli esempi sono fittizi e ogni riferimento a persone, società, indirizzi e-mail o URL esistenti è puramente casuale. Gli autori sono elencati nei documenti Riconoscimenti forniti insieme con questo software. I prodotti di terze parti e gli URL sono citati unicamente a scopo informativo e non costituiscono obbligo o raccomandazione. FileMaker, Inc. non si assume alcuna responsabilità nei confronti delle prestazioni di questi prodotti.

Per ulteriori informazioni, visitate il nostro sito Web all'indirizzo [www.filemaker.com/it](http://www.filemaker.com/it).

Edizione: 01

# Sommario

## Capitolo 1

### **Introduzione**

Informazioni su questa guida	7
Informazioni su ODBC e JDBC	7
Utilizzo del software FileMaker come applicazione client ODBC	8
Importazione dei dati ODBC	8
Aggiunta di tabelle ODBC al grafico delle relazioni	8
Utilizzo di un database FileMaker come origine dati	9
Accesso ad un database FileMaker Pro ospitato	9
Limitazioni con strumenti di terze parti	10
Requisiti di rete	10
Aggiornamento di file di versioni precedenti	10
Installazione dei driver correnti	10

## Capitolo 2

### **Accesso a origini dati SQL esterne**

Importazione dei dati ODBC	11
Eseguire SQL per interagire con le origini dati tramite ODBC	12
Uso delle tabelle ODBC nel grafico delle relazioni	13
Origini dati supportate in FileMaker 11	13
Aggiunta di tabelle ODBC al grafico delle relazioni	13

## Capitolo 3

### **Installazione dei driver client ODBC FileMaker**

Requisiti hardware e software	15
Requisiti per i driver client ODBC (Windows)	15
Requisiti per i driver client ODBC (Mac OS)	15
Requisiti di rete	15
Installazione del driver client ODBC (Windows)	15
Configurazione dei driver client (Windows)	16
Installazione del driver client ODBC (Mac OS)	17
Configurazione dei driver client (Mac OS)	18
Operazioni successive	19

## Capitolo 4

### **Uso di ODBC per condividere dati FileMaker**

Informazioni su ODBC	21
Uso del driver client ODBC	22
Accesso ad un file di database FileMaker	22
Accesso a un file di database FileMaker da un'applicazione Windows	23
Specifica delle proprietà del driver client ODBC per un DSN FileMaker (Windows)	23
Verifica dell'accesso attraverso ODBC (Windows)	24
Accesso ad un file di database FileMaker da un'applicazione Mac OS	25
Specifica delle proprietà del driver client ODBC per un DSN FileMaker (Mac OS)	25
Verifica dell'accesso attraverso ODBC (Mac OS)	26

## Capitolo 5

### **Installazione dei driver client JDBC di FileMaker**

Requisiti software	29
Requisiti di rete	29
Installazione del driver client JDBC	29
Uso del driver client JDBC	30

## Capitolo 6

### **Uso di JDBC per condividere dati FileMaker**

Informazioni su JDBC	31
Uso del driver client JDBC	31
Driver client JDBC	31
Uso di un URL JDBC per collegarsi al database	32
Specificazione delle proprietà del driver nel nome secondario dell'URL	34
Soluzioni con più file di database FileMaker	34
Verifica accesso tramite JDBC	35

## Capitolo 7

### **Standard supportati**

Supporto caratteri Unicode	37
Istruzioni SQL	37
Istruzione SELECT	37
clausole SQL	38
Clausola FROM	39
Clausola WHERE	39
Clausola GROUP BY	40
Clausola HAVING	40
Operatore UNION	40
Clausola ORDER BY	41
Clausola FOR UPDATE	41
Istruzione DELETE	44
Istruzione INSERT	44
Istruzione UPDATE	45
Istruzione CREATE TABLE	46
Istruzione ALTER TABLE	47
Istruzione CREATE INDEX	47
Istruzione DROP INDEX	48
Funzioni aggregate SQL	48
Espressioni SQL	49
Nomi di campo	49
Costanti	49
Notazione esponenziale/scientifica	50
Operatori numerici	50
Operatori alfabetici	50
Operatori data	51
Operatori relazionali	51
Operatori logici	52
Funzioni	53
Funzioni che restituiscono stringhe di caratteri	53
Funzioni che restituiscono numeri	54
Funzioni che restituiscono date	55
Precedenza degli operatori	56
Funzioni di Catalogo ODBC	56
Funzioni Meta Data JDBC	57
Parole chiave SQL riservate	57

## Capitolo 8

### **Informazioni di riferimento**

Mappatura di campi FileMaker su dati di tipo ODBC	61
Mappatura di campi FileMaker su dati di tipo JDBC	62
Messaggi di errore ODBC e JDBC	62
Messaggi di errore ODBC	62
Messaggi di errore JDBC	63
<b>Indice</b>	65

# Capitolo 1

## Introduzione

Questa guida descrive come utilizzare il software FileMaker® come applicazione client ODBC e come origine dati per applicazioni ODBC e JDBC.

La tabella seguente offre una panoramica su come utilizzare ODBC e JDBC con il software FileMaker.

Operazione da eseguire	Modalità di esecuzione	Prodotto	Vedere
<ul style="list-style-type: none"><li>Utilizzare il software FileMaker come applicazione client ODBC.</li><li>Accedere ai dati ODBC memorizzati in un'origine dati SQL esterna.</li></ul>	<ol style="list-style-type: none"><li>In modo interattivo attraverso il grafico delle relazioni</li><li>In modo statico mediante l'importazione ODBC o il menu File &gt; Apri. Inoltre, istruzione di script Importa record e istruzione di script Esegui SQL</li></ol>	<ul style="list-style-type: none"><li>FileMaker Pro</li><li>FileMaker Pro Advanced</li><li>FileMaker Server</li><li>FileMaker Server Advanced</li></ul>	<ul style="list-style-type: none"><li>Questa guida, capitolo 2.</li><li>Guida di FileMaker Pro</li></ul>
<ul style="list-style-type: none"><li>Utilizzare un database FileMaker come origine dati</li><li>Condividere i dati di FileMaker Pro con un'applicazione client ODBC di terze parti.</li></ul>	<ol style="list-style-type: none"><li>Query SQL</li><li>ODBC and JDBC</li></ol>	<ul style="list-style-type: none"><li>FileMaker Pro</li><li>FileMaker Pro Advanced</li><li>Solo FileMaker Server Advanced</li></ul>	Questa guida, capitoli da 3 a 8.

### Informazioni su questa guida

- Per informazioni sull'uso di ODBC e JDBC con le versioni precedenti di FileMaker Pro, visitare il sito [www.filemaker.com/it/support/index.html](http://www.filemaker.com/it/support/index.html).
- Per poter usare questa Guida è necessario avere conoscenze di base sull'uso di ODBC e JDBC e sulla costruzione di query SQL. Per informazioni su questi argomenti consultare il materiale di riferimento relativo.
- In questa guida il termine "FileMaker Pro" indica sia FileMaker Pro che FileMaker Pro Advanced, a meno che non vengano descritte caratteristiche specifiche di FileMaker Pro Advanced.

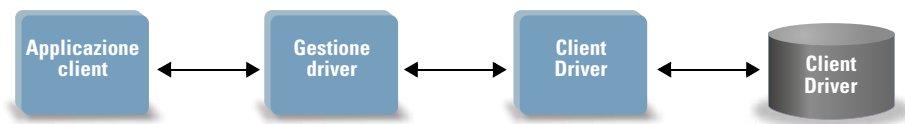
**Nota** È possibile scaricare la documentazione di FileMaker in formato PDF all'indirizzo [www.filemaker.com/it/support/index.html](http://www.filemaker.com/it/support/index.html). Eventuali aggiornamenti a questo documento sono disponibili anche sul sito web.

### Informazioni su ODBC e JDBC

ODBC e JDBC sono API (Application Programming Interfaces). ODBC è un'API per applicazioni scritte in linguaggio C, mentre JDBC è un'API analoga per il linguaggio Java. Queste API offrono alle applicazioni client un linguaggio comune per interagire con vari servizi di database e origini dati, compresi FileMaker Pro e FileMaker Server Advanced.

Tutte le applicazioni che supportano ODBC e JDBC riconoscono un sottogruppo fondamentale di istruzioni SQL (Structured Query Language). SQL consente di utilizzare altre applicazioni (fogli di calcolo, elaboratori di testi, strumenti per la generazione di resoconti) per visualizzare, analizzare e modificare i dati.

Utilizzando le API ODBC o JDBC, un'applicazione client comunica con una gestione driver che identifica il driver client per comunicare con una origine dati.



Il software FileMaker può agire o come applicazione client o come origine dati.

## Utilizzo del software FileMaker come applicazione client ODBC

Come applicazione client ODBC, il software FileMaker può accedere ai dati all'interno di origini dati SQL esterne. Il software FileMaker si connette all'origine dati SQL esterna utilizzando il driver client per l'origine dati ODBC e o importa dati ODBC o lavora con tabelle ODBC nel grafico delle relazioni.

### Importazione dei dati ODBC

È possibile importare i dati ODBC in uno dei seguenti modi:

- dal menu File, specificando un'origine dati ODBC e inserendo le istruzioni SQL nella finestra di dialogo dell'applicazione per la generazione di query SQL di FileMaker Pro
- creando uno script FileMaker che utilizza l'istruzione di script Importa record o l'istruzione di script Esegui SQL

Per ciascuno di questi metodi, è necessario inserire le istruzioni SQL da sé e quindi conoscere le istruzioni SQL supportate e la loro sintassi per l'origine dati ODBC. Dovendo inserire le istruzioni SQL da sé, è possibile importare i dati ODBC da qualsiasi origine dati ODBC.

### Aggiunta di tabelle ODBC al grafico delle relazioni

Quando si aggiunge una tabella ODBC al grafico delle relazioni, è possibile collegarsi e lavorare con i dati nelle origini dati SQL esterne in modo simile a come si lavora con i dati nel file di database FileMaker attualmente attivo. È ad esempio possibile

- creare le tabelle nel grafico delle relazioni per origini dati ODBC
- aggiungere campi supplementari a tabelle ODBC per effettuare calcoli o riepilogare dati nelle tabelle ODBC
- aggiungere, modificare ed eliminare i dati esterni in modo interattivo
- creare relazioni tra campi in tabelle e campi FileMaker (chiamati anche "colonne") nelle tabelle ODBC

Poiché le istruzioni SQL utilizzate per comunicare con la tabella ODBC aggiunta al grafico delle relazioni sono generate da FileMaker Pro, si è limitati a origini dati Oracle, SQL Server e MySQL specifiche, supportate da FileMaker Pro, nel grafico delle relazioni.

**Nota** Non è possibile modificare lo schema delle origini dati ODBC esterne utilizzando FileMaker Pro.

Il Capitolo 2, "Accesso a origini dati SQL esterne," descrive come utilizzare il software FileMaker come applicazione client ODBC.

## Utilizzo di un database FileMaker come origine dati

Come origine dati, i dati FileMaker sono condivisi con applicazioni compatibili con ODBC e JDBC. L'applicazione si collega all'origine dati FileMaker utilizzando il driver client FileMaker, genera ed esegue le query SQL utilizzando ODBC o JDBC ed elabora i dati recuperati dalla soluzione di database FileMaker.

### Accesso ad un database FileMaker Pro ospitato

FileMaker Server Advanced o FileMaker Pro consentono di ospitare un file di database FileMaker come origine dati, condividendo i dati con altre applicazioni usando ODBC e JDBC. La seguente tabella descrive le operazioni che possono essere effettuate con ogni prodotto FileMaker.

Questo prodotto FileMaker	Permette
FileMaker Server Advanced	Fino a 50 connessioni e supporta l'accesso locale (stesso computer) e l'accesso remoto (sia per middleware, come i server Web, sia per l'accesso remoto dei client da applicazioni produttive desktop).
FileMaker Pro	Fino a nove connessioni e supporta solo l'accesso locale (stesso computer).

Se la soluzione FileMaker utilizza più di un file di database, tutti i file devono essere sullo stesso computer.

I componenti plug-in ODBC e JDBC necessari per la condivisione dei dati con altre applicazioni sono installati con FileMaker Server Advanced e FileMaker Pro.

Per accedere ad un file di database FileMaker ospitato, installare il driver client ODBC o JDBC corrispondente. Installare il driver client sulla macchina in cui è installata l'applicazione di terze parti.

Questa guida spiega inoltre in che modo i driver client ODBC e JDBC, quando usati con FileMaker Pro e FileMaker Server Advanced, supportano gli standard industriali per ODBC (Open Database Connectivity), JDBC (Java Database Connectivity) e SQL (Structured Query Language).

- Il Capitolo 3, "Installazione dei driver client ODBC FileMaker," spiega come installare i file dei driver necessari per accedere all'origine dati FileMaker usando ODBC.
- Il Capitolo 4, "Uso di ODBC per condividere dati FileMaker," descrive come utilizzare il driver client ODBC FileMaker per connettersi ad un'origine dati FileMaker da un'altra applicazione.
- Il Capitolo 5, "Installazione dei driver client JDBC di FileMaker," spiega come installare i file dei driver necessari per accedere all'origine dati FileMaker usando JDBC.
- Il Capitolo 6, "Uso di JDBC per condividere dati FileMaker," descrive come utilizzare il driver client JDBC di FileMaker con un'applicazione Java o un applet che si collega a un'origine dati FileMaker.
- Il Capitolo 7, "Standard supportati," descrive le istruzioni SQL supportate dai driver client ODBC e JDBC quando vengono utilizzati con FileMaker Pro e FileMaker Server Advanced.

**Importante** Se si disattiva la condivisione ODBC/JDBC dopo averla attivata, l'origine dati ospitata da FileMaker Server Advanced o FileMaker Pro diventa immediatamente non disponibile. L'amministratore del database non ha la capacità di segnalare alle applicazioni client ODBC e JDBC la disponibilità dell'origine dati (l'amministratore può comunicare soltanto con i client del file di database FileMaker). Non viene segnalato nessun errore; l'applicazione client deve notificare agli utenti che l'origine dati non è più disponibile e che le transazioni non possono essere completate. Se un'applicazione client tenta di connettersi ad un file di database FileMaker non disponibile, viene visualizzato un messaggio che indica che la connessione è fallita.

## Limitazioni con strumenti di terze parti

Microsoft Access: Quando si utilizza Microsoft Access per visualizzare i dati in un'origine dati FileMaker, non utilizzare i dati di un campo Riassunto. I dati del campo Riassunto non devono essere modificati in Microsoft Access ed è possibile che il valore dati visualizzato in Microsoft Access non sia preciso.

## Requisiti di rete

Se si usa FileMaker Server Advanced per ospitare un file di database FileMaker come origine dati in una rete, è necessaria una rete TCP/IP. FileMaker Pro supporta soltanto l'accesso locale (sullo stesso computer).

## Aggiornamento di file di versioni precedenti

### Installazione dei driver correnti

Se è stato installato il driver di una versione precedente a FileMaker Pro o FileMaker Server Advanced, è necessario installare il driver per la versione 11.

Il driver per la versione 11 di FileMaker non è compatibile con versioni precedenti di FileMaker Pro o FileMaker Server Advanced.

Per ulteriori informazioni sull'installazione dei driver, vedere capitolo 3, "Installazione dei driver client ODBC FileMaker" e capitolo 5, "Installazione dei driver client JDBC di FileMaker."

**Nota** È necessario creare un DSN (Data Source Name) per ogni file di database FileMaker a cui si desidera accedere come origine dati. Se precedentemente è stato configurato l'accesso attraverso un DSN unico, che consente alle tabelle di suddividersi in diversi file di database FileMaker, sarà necessario consolidare queste tabelle in un unico file di database (oppure creare diversi DSN).

## Capitolo 2

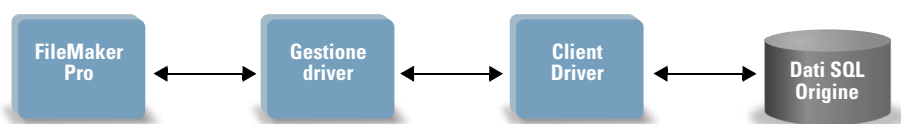
# Accesso a origini dati SQL esterne

Come applicazione client ODBC, il software FileMaker può accedere ai dati all'interno di origini dati SQL esterne. Il software FileMaker si connette all'origine dati SQL esterna utilizzando il driver client per l'origine dati ODBC e o importa dati ODBC o lavora con tabelle ODBC nel grafico delle relazioni.

Se si stanno importando dati ODBC o si sta lavorando con tabelle ODBC nel grafico delle relazioni, è necessario configurare un driver per l'origine dati ODBC che si sta usando. Ad esempio, per accedere ai record da un database Oracle è necessario configurare un driver client Oracle.

### Importazione dei dati ODBC

Quando si importano i dati ODBC, è necessario un driver client ODBC configurato per l'origine dati SQL esterna sulla macchina client.



Dopo aver configurato un driver client ODBC, è possibile interagire con i record, importarli in un file di database FileMaker Pro esistente o creare un nuovo file di database FileMaker Pro da un'origine dati ODBC (come database di Oracle e Microsoft Access).

Per prima cosa bisogna accedere all'origine dati da importare. Quindi si costruisce una query per i record da importare dall'origine dati. Infine, se si stanno importando dati in un file esistente, si cerca di far corrispondere i campi dell'origine dati con quelli nel file del database di FileMaker Pro.

È possibile accedere all'origine dati ODBC attraverso il menu File, con l'istruzione di script **Importa record** o con l'istruzione di script **Esegui SQL**.

Per importare i dati ODBC, seguire questa procedura generale:

- Installare e configurare i driver ODBC specifici per le origini dati esterne a cui si desidera accedere.
- Sul computer che ospita il file FileMaker Pro corrente, definire un Nome Origine Dati (DSN) di sistema per ogni origine dati ODBC a cui si desidera accedere.
- Determinare qualsiasi considerazione aggiuntiva per le origini dati ODBC a cui si desidera accedere (ad esempio, se agli utenti viene richiesto di indicare nome utente e password).

- In FileMaker Pro, eseguire una delle seguenti operazioni:
  - Per importare in un file FileMaker Pro esistente, selezionare il menu File > Importa record > Origine dati ODBC.

- Per creare un file FileMaker Pro dai record dell'origine dati, selezionare il menu File > Apri. Nella finestra di dialogo Apri, selezionare Origine dati ODBC in Tipo file (Windows) o Mostra (Mac OS).

Selezionare l'origine dati, inserire nome utente e password (se presente) e fare clic su OK per aprire la finestra di dialogo dell'applicazione per la generazione di query SQL fornita con FileMaker Pro.

Utilizzando la finestra di dialogo dell'applicazione per la generazione di query SQL fornita con FileMaker Pro è possibile creare una query. Selezionare la tabella da cui si desidera eseguire l'importazione, quindi selezionare le colonne specifiche che si desidera utilizzare nella query SQL. Utilizzare la scheda WHERE per creare i criteri di ricerca e la scheda ORDER BY per specificare un criterio di ordinamento.

È anche possibile digitare un'istruzione SQL direttamente nella finestra di dialogo dell'applicazione per la generazione di query SQL.

È possibile eseguire immediatamente la query oppure utilizzare l'istruzione di script Importa record o Esegui SQL per eseguire una query come parte di uno script FileMaker.

**Nota** L'importazione ODBC, l'istruzione di script SQL e le origini dati SQL esterne non sono supportate nelle soluzioni runtime create con FileMaker Pro Advanced.

Per ulteriori informazioni sull'importazione dei dati, sull'utilizzo della finestra di dialogo dell'applicazione per la generazione di query SQL e sulla creazione di script FileMaker consultare la Guida di FileMaker Pro.

## Eseguire SQL per interagire con le origini dati tramite ODBC

Oltre a importare dati in un file di database FileMaker Pro attraverso ODBC, è anche possibile interagire con le origini dati utilizzando le istruzioni SQL attraverso l'istruzione di script Esegui SQL. È possibile utilizzare qualsiasi istruzione SQL supportata dalle origini dati, quali INSERT, UPDATE e DELETE.

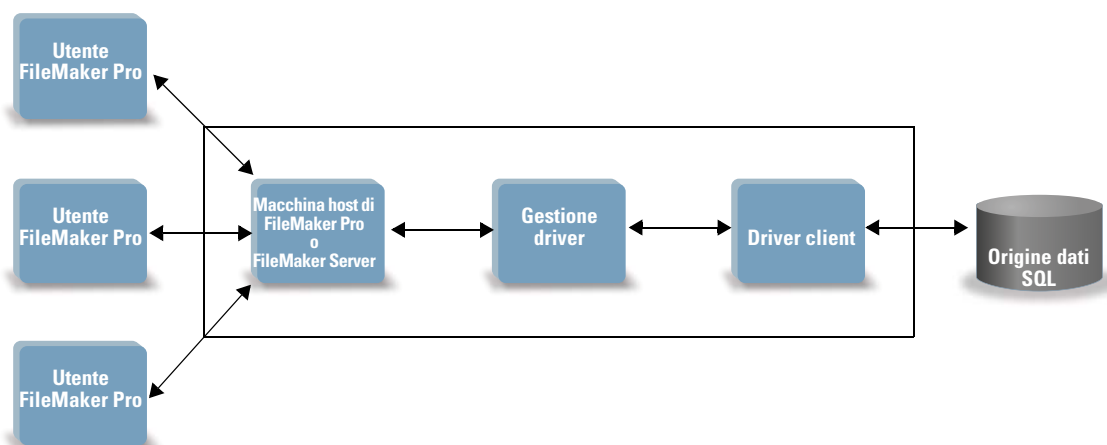
È anche possibile utilizzare istruzioni SQL che vanno oltre la semplice importazione di dati in un file di database FileMaker Pro. Ad esempio, si potrebbero eseguire istruzioni SQL che aggiungono record a una tabella di un database nel Server SQL, utilizzando informazioni da un file di database FileMaker Pro.

Consultare la Guida di FileMaker Pro per ulteriori informazioni su come creare script FileMaker che usano l'istruzione di script Esegui SQL.

## Uso delle tabelle ODBC nel grafico delle relazioni

Quando si aggiunge una tabella ODBC al grafico delle relazioni, è possibile collegarsi e lavorare con i dati nelle origini dati SQL esterne in modo simile a come si lavora con i dati nel file di database FileMaker attualmente attivo.

Quando si usa FileMaker Pro o Filemaker Server come host per una soluzione che comprende le tabelle ODBC nel grafico delle relazioni, si deve configurare il driver client ODBC per l'origine dati SQL esterna sulla macchina host.



### Origini dati supportate in FileMaker 11

In qualità di applicazione client ODBC, FileMaker supporta le seguenti origini dati SQL esterne come tabelle ODBC nel grafico delle relazioni:

- Oracle 9 i
- Oracle 10 g
- Oracle 11 g
- SQL Server 2000
- SQL Server 2005
- SQL Server 2008
- MySQL 5.0 Community Edition (gratuito)
- MySQL 5,1 Community Edition (gratuito)

Per informazioni sui driver client supportati consultare [www.filemaker.com/it/support](http://www.filemaker.com/it/support).

### Aggiunta di tabelle ODBC al grafico delle relazioni

Per impostare un database FileMaker Pro per l'accesso ai dati nelle origini dati ODBC supportate:

- Installare e configurare i driver ODBC specifici per le origini dati esterne a cui si desidera accedere.
- Sul computer che ospita il file FileMaker Pro corrente, definire un Nome Origine Dati (DSN) di sistema per ogni origine dati ODBC a cui si desidera accedere.
- Determinare qualsiasi considerazione aggiuntiva per le origini dati ODBC a cui si desidera accedere (ad esempio, se agli utenti viene richiesto di indicare nome utente e password).

- Aggiungere una o più tabelle dall'origine dati ODBC al grafico delle relazioni nel file FileMaker Pro corrente.
- Per visualizzare dati esterni, aggiungere i campi ai formati del file FileMaker Pro.
- Se si desidera, aggiungere campi supplementari alle tabelle e ai formati esterni per visualizzare i risultati di calcolo e riassunto in base ai dati memorizzati nelle origini dati ODBC esterne.

Consultare la Guida di FileMaker Pro per la descrizione dettagliata dei passi e per ulteriori informazioni sulla configurazione di un driver client ODBC, sulla connessione alle origini dati ODBC, sulla modifica delle origini dati ODBC e sull'impostazione di una tabella ODBC nel grafico delle relazioni.

# Capitolo 3

## ***Installazione dei driver client ODBC FileMaker***

Queste istruzioni spiegano come installare il driver client ODBC necessario per accedere a FileMaker come origine dati da applicazioni di terze parti e da applicazioni personalizzate attraverso ODBC (Open Database Connectivity). Il driver client ODBC è disponibile nella cartella xDBC tramite un'installazione separata sul disco di installazione di FileMaker o scaricandolo da Internet.

Le ultime versioni dei driver client sono disponibili anche in questo URL:

[www.filemaker.com/it/support](http://www.filemaker.com/it/support)

Se il file di database FileMaker verrà ospitato usando FileMaker Server Advanced, rendere i driver client disponibili per gli utenti remoti.

Dopo che il driver client necessario è stato installato, è possibile configurarlo per accedere all'origine dati FileMaker e costruire query SQL (Structured Query Language) per interagire con i dati.

### **Requisiti hardware e software**

Per installare e utilizzare i driver client ODBC, sono necessari i seguenti requisiti minimi hardware e software:

#### **Requisiti per i driver client ODBC (Windows)**

- Pentium III 700MHz o superiore
- 256 MB di RAM minimo, 2 GB di RAM raccomandato per Windows XP; 1 GB di RAM minimo per Windows Vista e Windows 7
- Microsoft Data Access Components (MDAC) 2.8 SP1
- Windows MDAC 6.0 per Vista

#### **Requisiti per i driver client ODBC (Mac OS)**

- Apple G4 o G5; o computer Mac OS con processore Intel
- 1 GB di RAM minimo; 2 GB di RAM raccomandato
- Mac OS X, versione 10.5.7 o 10.6 (il software può anche essere utilizzato con le versioni successive certificate da FileMaker)

#### **Requisiti di rete**

Se si accede ad un'origine dati FileMaker ospitata su un computer diverso, è necessario accedere alla rete mediante il protocollo TCP/IP.

### **Installazione del driver client ODBC (Windows)**

**Nota** Per installare il driver client ODBC su un sistema operativo Windows a 32 bit è necessario MDAC 2.8 SP1 (disponibile sul sito Internet [www.microsoft.com](http://www.microsoft.com)).

### Installazione del driver client ODBC:

1. Eseguire una delle seguenti operazioni:
  - Se il software è stato scaricato elettronicamente, fare doppio clic sull'icona di installazione (file .exe).
  - Se si dispone di un disco di installazione, inserirlo nell'apposita unità.
2. Nella finestra FileMaker Pro o Filemaker Server, fare doppio clic sulla cartella xDBC.
3. Nella cartella xDBC, fare doppio clic sulla cartella ODBC Client Driver Installer.
4. Nella cartella ODBC Client Driver Installer, fare doppio clic sul file FMODBC\_Installer\_Win32.msi.  
Si apre la finestra dell'Installazione guidata del driver ODBC di FileMaker.
5. Installare il driver client ODBC seguendo le istruzioni visualizzate sullo schermo.
6. Al termine dell'installazione, fare clic su **Chiudi**.

Per impostazione predefinita il driver client ODBC viene installato nella seguente cartella:

c:\windows\system32

Il driver client ODBC, ODBC FileMaker, è ora disponibile; permette di configurare l'accesso a un'origine dati FileMaker.

## Configurazione dei driver client (Windows)

Prima di poter usare un'applicazione client per accedere ad un'origine dati di FileMaker Pro è necessario configurare un driver client per l'origine dati. Le impostazioni di configurazione indicano il driver client in uso, la posizione dell'origine dati e i dettagli relativi al metodo di connessione che si intende utilizzare.

**Importante** Quando si utilizza un driver client FileMaker, è necessario specificare 2399 come porta.

### Per configurare il driver client ODBC:

1. Nel pannello di controllo di Windows, aprire **Strumenti di amministrazione > Origini dati (ODBC)**.
  - In Windows XP, gli strumenti di amministrazione appaiono nella categoria **Prestazioni e manutenzione**.
  - In Windows Vista e Windows 7, Strumenti di amministrazione appare nella categoria **Sistema e sicurezza**.Si apre l'Amministratore Origine dati ODBC.
2. Selezionare la scheda **DSN di sistema** o **DSN Utente**
3. Fare clic su **Aggiungi**.  
Si apre la finestra di dialogo **Crea origine dati**.
4. Selezionare **ODBC FileMaker** e fare clic su **Fine**.  
Si apre la finestra di dialogo **Configurazione FileMaker DSN**.
5. Fare clic su **Avanti**.
6. Per **Nome** digitare un nome che sia significativo per chi accede all'origine dati FileMaker.  
Per **Descrizione** inserire un'eventuale descrizione dell'origine dati FileMaker. Fare clic su **Avanti**.

**7. Per Host:**

- Se si esegue la connessione a un file di database ospitato da FileMaker Pro sul proprio computer locale, inserire localhost o l'indirizzo IP 127.0.0.1.
- Se si esegue la connessione ad un file di database ospitato da FileMaker Server Advanced in rete, inserire l'indirizzo IP di FileMaker Server.

Se è stata attivata la condivisione mediante ODBC/JDBC nell'applicazione host, è possibile selezionare **Connettersi all'host per ottenere i nomi dei database disponibili**. Fare clic su **Avanti**.

In caso contrario fare clic su **Fine** per salvare le informazioni sull'origine dati.

**8. Per Database** selezionare un database dalla lista dei database disponibili o digitare il nome del file di database FileMaker che si sta usando come origine dati.

**Nota** Per i file di database ospitati da FileMaker Server Advanced, la lista dei database può essere filtrata in base all'impostazione **Filtro file visualizzati**. Per informazioni vedere la Guida di FileMaker Server.

Se è necessaria la gestione speciale di testo non inglese, fare clic su **Lingue avanzate**. Si apre la finestra di dialogo **Opzioni Lingue avanzate**.

- Per rilevare automaticamente le impostazioni della lingua, selezionare l'opzione **Rileva automaticamente impostazioni lingua per l'applicazione**.
- Per specificare l'impostazione della lingua, deselezionare l'opzione **Rileva automaticamente impostazioni lingua per l'applicazione** e selezionare l'impostazione di sistema che si desidera usare.

Per creare un file di registro per le query a esecuzione lunga, selezionare l'opzione **Salva query a esecuzione lunga in file di registro** e digitare un nome per il file di registro.

Fare clic su **Fine** per salvare le informazioni sull'origine dati.

**9. Esaminare le informazioni sul DSN FileMaker.**

- Fare clic su **Test** per verificare che il driver client ODBC sia stato configurato correttamente per accedere all'origine dati FileMaker.

Se si riceve un messaggio di errore, è possibile correggere le informazioni sulla connessione.

Può essere inoltre necessario controllare che il file di database FileMaker sia ospitato e disponibile, che l'account FileMaker specificato utilizzi un set di privilegi con il privilegio esteso di **Accesso tramite ODBC/JDBC** e che l'applicazione host (FileMaker Pro o FileMaker Server Advanced) sia stata predisposta per la condivisione attraverso ODBC/JDBC.

- Fare clic su **OK** per salvare le informazioni sull'origine dati.

## Installazione del driver client ODBC (Mac OS)

### Installazione del driver client ODBC:

**1. Eseguire una delle seguenti operazioni:**

- Se il software è stato scaricato elettronicamente, fare doppio clic sull'icona con l'immagine del disco (file .dmg).
- Se si dispone di un disco di installazione, inserirlo nell'apposita unità.

2. Nella finestra FileMaker Pro o Filemaker Server, fare doppio clic sulla cartella xDBC.
3. Nella cartella xDBC, fare doppio clic sulla cartella ODBC Client Driver Installer.
4. Nella cartella ODBC Client Driver Installer, fare doppio clic su FileMaker ODBC.mpkg.  
Si apre la finestra FileMaker ODBC Driver Installer.
5. Installare il driver client ODBC seguendo le istruzioni visualizzate sullo schermo.
6. Al termine dell'installazione, fare clic su Chiudi.

Il driver client ODBC viene installato nella seguente cartella:  
/Libreria/ODBC

**Nota** Non è possibile cambiare la cartella di installazione del driver client ODBC.

Il driver client ODBC, ODBC FileMaker, è ora disponibile; permette di configurare l'accesso a un'origine dati FileMaker.

## Configurazione dei driver client (Mac OS)

Prima di poter usare un'applicazione client per accedere ad un'origine dati di FileMaker Pro è necessario configurare un driver client per l'origine dati. Le impostazioni di configurazione indicano il driver client in uso, la posizione dell'origine dati e i dettagli relativi al metodo di connessione che si intende utilizzare.

Queste istruzioni presuppongono l'installazione dell'ODBC Manager di Actual Technologies, disponibile nel sito [www.odbcmanager.net](http://www.odbcmanager.net), che è un prodotto freeware non supportato da FileMaker.

Si può utilizzare anche lo strumento Amministratore ODBC di Apple per Mac OS X. Per Mac OS X versione 10.5, l'Amministratore ODBC è installato come parte del sistema operativo. Per Mac OS X versione 10.6, l'Amministratore ODBC è disponibile sotto [www.apple.com/support](http://www.apple.com/support).

**Importante** Quando si utilizza un driver client FileMaker, è necessario specificare 2399 come porta.

### Per configurare il driver client ODBC:

1. Avviare l'utilità ODBC Manager. (ODBC Manager è installato nella cartella Utilità nella cartella Applicazioni).
2. Selezionare la scheda DSN di sistema o DSN Utente e fare clic su Aggiungi.  
Si apre la finestra di dialogo Scegli un driver.
3. Selezionare ODBC FileMaker e fare clic su OK.  
Si apre la finestra di dialogo Configurazione FileMaker DSN.
4. Fare clic su Continua.
5. Per Nome digitare un nome che sia significativo per chi accede all'origine dati FileMaker.  
Per Descrizione inserire, se lo si desidera, una descrizione dell'origine dati FileMaker. Fare clic su Continua.

**6. Per Host:**

- Se si esegue la connessione a un file di database ospitato da FileMaker Pro sul proprio computer locale, inserire localhost o l'indirizzo IP 127.0.0.1.
- Se si esegue la connessione ad un file di database ospitato da FileMaker Server Advanced in rete, inserire l'indirizzo IP di FileMaker Server.

Se è stata attivata la condivisione mediante ODBC/JDBC nell'applicazione host, è possibile selezionare **Connettersi all'host per ottenere i nomi dei database disponibili**. Fare clic su **Continua**.

In caso contrario fare clic su **Fine** per salvare le informazioni sull'origine dati.

**7. Per Database** selezionare un database dalla lista dei database disponibili o digitare il nome del file di database FileMaker che si sta usando come origine dati.

**Nota** Per i file di database ospitati da FileMaker Server Advanced, la lista dei database può essere filtrata in base all'impostazione **Filtro file visualizzati**. Per informazioni vedere la Guida di FileMaker Server.

Se è necessaria la gestione speciale di testo non inglese, fare clic su **Lingue avanzate**. Si apre la finestra di dialogo **Opzioni Lingue avanzate**.

- Per rilevare automaticamente le impostazioni della lingua, selezionare l'opzione **Rileva automaticamente impostazioni lingua per l'applicazione**.
- Per specificare l'impostazione della lingua, deselezionare l'opzione **Rileva automaticamente impostazioni lingua per l'applicazione** e selezionare l'impostazione di sistema che si desidera usare.

Per creare un file di registro per le query a esecuzione lunga, selezionare l'opzione **Salva query a esecuzione lunga in file di registro** e digitare un nome per il file di registro.

Fare clic su **Fine** per salvare le informazioni sull'origine dati.

**8. Esaminare le informazioni sul DSN FileMaker.**

- Fare clic su **Test** per verificare che il driver client ODBC sia stato configurato correttamente per accedere all'origine dati FileMaker.

Se si riceve un messaggio di errore, è possibile correggere le informazioni sulla connessione.

Può essere inoltre necessario controllare che il file di database FileMaker sia ospitato e disponibile, che l'account FileMaker specificato utilizzi un set di privilegi con il privilegio esteso di **Accesso tramite ODBC/JDBC** e che l'applicazione host (FileMaker Pro o FileMaker Server Advanced) sia stata predisposta per la condivisione attraverso ODBC/JDBC.

- Fare clic su **OK** per salvare le informazioni sull'origine dati.

## Operazioni successive

Dopo aver installato e configurato un driver client è possibile costruire ed eseguire query SQL per accedere all'origine dati FileMaker.

Talvolta le applicazioni client usano una terminologia diversa per accedere ad un'origine dati tramite ODBC. In molte applicazioni vi sono voci di menu quali **Get external data** o **SQL query**. Per ulteriori dettagli consultare la documentazione o la Guida fornite con l'applicazione.

Per ulteriori informazioni sull'utilizzo di FileMaker come un'origine dati ODBC, vedere capitolo 4, "Uso di ODBC per condividere dati FileMaker."



# Capitolo 4

## Uso di ODBC per condividere dati FileMaker

Utilizzare il driver client ODBC per collegarsi a un'origine dati FileMaker da un'altra applicazione. L'applicazione che usa il driver client ODBC può accedere direttamente ai dati in un file di database FileMaker. Il driver client ODBC FileMaker è ODBC FileMaker.

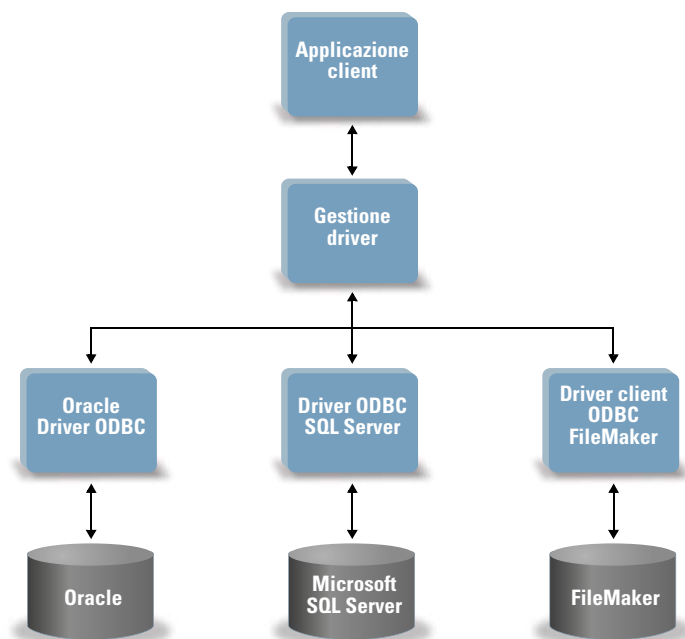
**Nota** È inoltre possibile utilizzare FileMaker Pro come applicazione client ODBC che interagisce con i record di un'altra origine dati tramite ODBC usando SQL. Per ulteriori informazioni sull'accesso a origini dati esterne SQL tramite ODBC vedere capitolo 2, "Accesso a origini dati SQL esterne,".

### Informazioni su ODBC

ODBC è un'API che consente alle applicazioni di accedere ai dati da diversi sistemi di gestione di database. ODBC offre alle applicazioni client un linguaggio comune per interagire con servizi di database e origini dati.

Tutte le applicazioni che supportano ODBC riconoscono un sottogruppo fondamentale di istruzioni SQL (Structured Query Language). SQL permette di usare altre applicazioni (come fogli di calcolo, elaboratori di testo e strumenti per la generazione di resoconti) per visualizzare, analizzare e modificare i dati FileMaker. Per le espressioni, le funzioni e le istruzioni SQL supportate dal driver client ODBC vedere la sezione capitolo 7, "Standard supportati,".

L'applicazione può interagire direttamente con un file di database FileMaker utilizzando il driver client ODBC. Le istruzioni SQL vengono trasmesse all'host del file di database FileMaker e i risultati di tali istruzioni vengono rispediti al mittente. Se si usa FileMaker Server Advanced per ospitare un file di database FileMaker come origine dati, il file di database può trovarsi su un'altra macchina (la macchina server) collegata alla rete, mentre l'applicazione client si trova sul proprio computer (la macchina client). Questa relazione è detta configurazione client/server.



## Uso del driver client ODBC

È possibile utilizzare il driver client ODBC con qualsiasi applicazione compatibile ODBC. Condividendo il file di database FileMaker come origine dati, è possibile:

- eseguire stampe unione con Microsoft Word
- creare grafici con Microsoft Excel
- spostare dati FileMaker in un DBMS, come Microsoft SQL Server
- analizzare ulteriormente i dati di FileMaker Pro con strumenti per la generazione di query o resoconti, per creare grafici, generare query mirate ed eseguire analisi specifiche
- creare un'applicazione Microsoft Visual Basic in grado di condividere informazioni con FileMaker Pro

Per condividere un file di database FileMaker come origine dati, utilizzare FileMaker Pro per definire gli account che devono avere accesso al file di database. Quindi, controllare l'accesso al file di database assegnando set di privilegi agli account, compreso il privilegio esteso di accesso attraverso ODBC/JDBC. Infine, attivare la condivisione dei dati mediante ODBC/JDBC nell'applicazione host FileMaker Server Advanced o FileMaker Pro. Per ulteriori dettagli vedere la Guida di FileMaker Pro o FileMaker Server.

**Importante** Le versioni precedenti del driver client ODBC FileMaker non sono compatibili con la versione 11 di FileMaker. Per collegarsi a un file di database della versione 11 di FileMaker, è necessario installare e configurare il nuovo driver client ODBC.

**Nota** Per verificare se FileMaker xDBC Listener è in esecuzione, è possibile utilizzare il Pannello di controllo (Mac OS) oppure il Task Manager (Windows) per controllare lo stato del processo FileMaker xDBC Listener. Quando questo processo viene avviato, gli viene assegnato il nome fmxdbc\_listener, utilizzato anche per registrare gli eventi. Il processo FileMaker xDBC Listener è distinto dal processo FileMaker Server.

## Accesso ad un file di database FileMaker

Da un'applicazione compatibile ODBC è possibile creare query SQL per accedere ad un file di database FileMaker. Il driver client ODBC deve essere installato sul computer che genera la query SQL.

Per accedere ad un file di database FileMaker:

1. In FileMaker Pro, esaminare il set di privilegi assegnato agli account che avranno accesso al file di database. Gli account a cui deve essere consentito l'accesso necessitano di un set di privilegi con il privilegio esteso Accesso tramite ODBC/JDBC.
2. Attivare la condivisione dei dati tramite ODBC/JDBC nell'applicazione host di FileMaker Server Advanced (mediante Admin Console di FileMaker Server) o FileMaker Pro.
 

Admin Console di FileMaker Server: Fare clic su ODBC/JDBC quindi selezionare Attiva ODBC/JDBC.

FileMaker Pro: Selezionare il menu File > Condivisione > ODBC/JDBC e impostare Condivisione ODBC/JDBC su Attivata.
3. Assicurarsi che il file di database FileMaker a cui si desidera accedere sia ospitato e disponibile.
 

Se la soluzione del database FileMaker utilizza più di un file del database, tutti i file devono essere sullo stesso computer.

4. Connettersi all'origine dati FileMaker.
5. Creare ed eseguire una query SQL nell'applicazione client.

Ogni file di database FileMaker aperto e configurato per l'accesso è un'origine dati separata (si crea un DSN per ogni file di database FileMaker a cui si desidera accedere come origine dati).

Ogni database può avere una o più tabelle. I campi FileMaker sono rappresentati sotto forma di colonne. Il nome completo del campo, compresi gli eventuali caratteri non alfanumerici, viene visualizzato come nome di colonna.

## Accesso a un file di database FileMaker da un'applicazione Windows

### Specifiche delle proprietà del driver client ODBC per un DSN FileMaker (Windows)

Creare un DSN per ogni file di database FileMaker a cui si desidera accedere come origine dati. Il DSN identifica il driver client ODBC FileMaker, la posizione dell'applicazione host e il file di database FileMaker a cui si sta accedendo come origine dati.

Per configurare un nuovo driver client ODBC, vedere "Configurazione dei driver client (Windows)" a pagina 16.

Per modificare un driver client ODBC esistente:

1. Nel pannello di controllo di Windows, aprire Strumenti di amministrazione > Origini dati (ODBC).
  - In Windows XP, gli strumenti di amministrazione appaiono nella categoria Prestazioni e manutenzione.
  - In Windows Vista e Windows 7, Strumenti di amministrazione appare nella categoria Sistema e sicurezza.
 Si apre l'Amministratore Origine dati ODBC.
2. Selezionare la scheda DSN di sistema o DSN Utente (selezionare la scheda utilizzata durante la configurazione).
3. Scegliere l'origine dati FileMaker configurata.
 

Il nome dell'origine dati inserito originariamente verrà visualizzato sotto Nome e ODBC FileMaker sarà il Driver.
4. Fare clic su Configura.
 

Si apre la finestra di dialogo Configurazione FileMaker DSN.
5. Per Nome digitare un nome che sia significativo per chi accede all'origine dati FileMaker.
 

Per Descrizione inserire un'eventuale descrizione dell'origine dati FileMaker. Fare clic su Avanti.
6. Per Host, inserire la posizione dell'origine dati.
 

Se si esegue la connessione ad un file di database FileMaker ospitato da FileMaker Pro sul proprio computer locale, digitare localhost (oppure 127.0.0.1).

Se si esegue la connessione ad un file di database FileMaker ospitato da FileMaker Server Advanced in rete, digitare l'indirizzo IP di FileMaker Server.

Se è stata attivata la condivisione mediante ODBC/JDBC nell'applicazione host, selezionare Connettersi all'host per ottenere i nomi dei database disponibili.

7. Per **Database** selezionare un database dalla lista dei database disponibili o digitare il nome del file di database FileMaker che si sta usando come origine dati.

**Nota** Per i file di database ospitati da FileMaker Server Advanced, la lista dei database può essere filtrata in base all'impostazione **Filtro file visualizzati**. Per informazioni vedere la Guida di FileMaker Server.

Se è necessaria la gestione speciale di testo non inglese, fare clic su **Lingue avanzate**. Si apre la finestra di dialogo **Opzioni Lingue avanzate**.

- Per rilevare automaticamente le impostazioni della lingua, selezionare l'opzione **Rileva automaticamente impostazioni lingua per l'applicazione**.
- Per specificare l'impostazione della lingua, deselezionare l'opzione **Rileva automaticamente impostazioni lingua per l'applicazione** e selezionare l'impostazione di sistema che si desidera usare.

Per creare un file di registro per le query a esecuzione lunga, selezionare l'opzione **Salva query a esecuzione lunga in file di registro** e digitare un nome per il file di registro.

8. Fare clic su **Fine** per salvare le informazioni sull'origine dati.
9. Fare clic su **OK** per chiudere la finestra di dialogo **Configurazione FileMaker DSN**.

### Verifica dell'accesso attraverso ODBC (Windows)

Per verificare che il driver client ODBC sia stato configurato correttamente per l'accesso all'origine dati FileMaker:

1. Nel pannello di controllo di Windows, aprire **Strumenti di amministrazione > Origini dati (ODBC)**.
  - In Windows XP, gli strumenti di amministrazione appaiono nella categoria **Prestazioni e manutenzione**.
  - In Windows Vista e Windows 7, Strumenti di amministrazione appare nella categoria **Sistema e sicurezza**.Si apre l'**Amministratore Origine dati ODBC**.
2. Selezionare la scheda **DSN di sistema** o **DSN Utente** (selezionare la scheda utilizzata durante la configurazione).
3. Scegliere l'origine dati FileMaker configurata.

Il nome dell'origine dati inserito originariamente verrà visualizzato sotto **Nome** e **ODBC FileMaker** sarà il **Driver**.
4. Fare clic su **Configura**.

Si apre la finestra di dialogo **Configurazione FileMaker DSN**.
5. Fare clic su **Avanti** finché non si arriva alla pagina **Conclusione**.
6. Fare clic su **Test**.

Viene richiesto di inserire il nome utente FileMaker (in **Nome utente database**) e la password (in **Password database**).

Se la connessione è OK, viene visualizzato il messaggio **Test completato**.

Se la connessione fallisce:

- Assicurarsi che il file di database FileMaker sia ospitato e disponibile.

- Aggiornare o correggere le informazioni sulla connessione.
- Assicurarci che il set di privilegi usi un account FileMaker con il privilegio esteso di Accesso tramite ODBC/JDBC.
- Verificare che per l'applicazione host FileMaker Pro o FileMaker Server sia stata configurata la condivisione tramite ODBC/JDBC.

## Accesso ad un file di database FileMaker da un'applicazione Mac OS

### Specifiche delle proprietà del driver client ODBC per un DSN FileMaker (Mac OS)

Creare un DSN per ogni file di database FileMaker a cui si desidera accedere come origine dati. Il DSN identifica il driver client ODBC FileMaker, la posizione dell'applicazione host e il file di database FileMaker a cui si sta accedendo come origine dati.

Per configurare un nuovo driver client ODBC, vedere "Configurazione dei driver client (Mac OS)" a pagina 18.

Per modificare un driver client ODBC esistente:

**1.** Avviare l'utilità ODBC Manager. (ODBC Manager è installato nella cartella Utilità nella cartella Applicazioni).

**2.** Fare clic sulla scheda DSN di sistema o DSN Utente.

**3.** Scegliere l'origine dati FileMaker configurata.

Il nome dell'origine dati inserito originariamente verrà visualizzato sotto Nome e ODBC FileMaker sarà il Driver.

**4.** Fare clic su Configura.

Si apre la finestra di dialogo Configurazione FileMaker DSN.

**5.** Fare clic su Continua.

**6.** Per Nome digitare un nome che sia significativo per chi accede all'origine dati FileMaker.

Un'ulteriore Descrizione è facoltativa.

**7.** Per Host, inserire la posizione dell'origine dati.

Se si esegue la connessione ad un file di database FileMaker ospitato da FileMaker Pro sul proprio computer locale, digitare localhost (oppure 127.0.0.1).

Se si esegue la connessione ad un file di database FileMaker ospitato da FileMaker Server Advanced in rete, digitare l'indirizzo IP di FileMaker Server.

Se è stata attivata la condivisione mediante ODBC/JDBC nell'applicazione host, selezionare Connettersi all'host per ottenere i nomi dei database disponibili.

8. Per **Database** selezionare un database dalla lista dei database disponibili o digitare il nome del file di database FileMaker che si sta usando come origine dati.

**Nota** Per i file di database ospitati da FileMaker Server Advanced, la lista dei database può essere filtrata in base all'impostazione **Filtro file visualizzati**. Per informazioni vedere la Guida di FileMaker Server.

Se è necessaria la gestione speciale di testo non inglese, fare clic su **Lingue avanzate**. Si apre la finestra di dialogo **Opzioni Lingue avanzate**.

- Per rilevare automaticamente le impostazioni della lingua, selezionare l'opzione **Rileva automaticamente impostazioni lingua per l'applicazione**.
- Per specificare l'impostazione della lingua, deselezionare l'opzione **Rileva automaticamente impostazioni lingua per l'applicazione** e selezionare l'impostazione di sistema che si desidera usare.

Per creare un file di registro per le query a esecuzione lunga, selezionare l'opzione **Salva query a esecuzione lunga in file di registro** e digitare un nome per il file di registro.

9. Fare clic su **Fine** per salvare le informazioni sull'origine dati.
10. Fare clic su **OK** per chiudere la finestra di dialogo **Configurazione FileMaker DSN**.

### Verifica dell'accesso attraverso ODBC (Mac OS)

Per verificare che il driver client ODBC sia stato configurato correttamente per l'accesso all'origine dati FileMaker:

1. Avviare l'utilità **ODBC Manager**. (**ODBC Manager** è installato nella cartella **Utilità** nella cartella **Applicazioni**).
2. Selezionare la scheda **DSN di sistema** o **DSN Utente** (selezionare la scheda utilizzata durante la configurazione).
3. Scegliere l'origine dati FileMaker configurata.  
Il nome dell'origine dati inserito originariamente verrà visualizzato sotto **Nome** e **ODBC FileMaker** sarà il **Driver**.
4. Fare clic su **Configura**.  
Si apre la finestra di dialogo **Configurazione FileMaker DSN**.
5. Fare clic su **Continua** fino ad arrivare alla pagina **Conclusione**.
6. Fare clic su **Test**.  
Viene richiesto di inserire il nome utente FileMaker (in **Nome utente database**) e la password (in **Password database**).

Se la connessione è OK, viene visualizzato il messaggio **Test completato**.

Se la connessione fallisce:

- Assicurarsi che il file di database FileMaker sia ospitato e disponibile.
- Aggiornare o correggere le informazioni sulla connessione.

- Assicurarsi che il set di privilegi usi un account FileMaker con il privilegio esteso di **Accesso tramite ODBC/JDBC**.
- Verificare che per l'applicazione host FileMaker Pro o FileMaker Server sia stata configurata la condivisione tramite ODBC/JDBC.



# Capitolo 5

## ***Installazione dei driver client JDBC di FileMaker***

Queste istruzioni spiegano come installare il driver client necessario per accedere a FileMaker come origine dati da applicazioni di terze parti e da applicazioni personalizzate attraverso JDBC (Java Database Connectivity). Il driver client è disponibile nella cartella xDBC o sul DVD FileMaker o scaricandolo da Internet. Le ultime versioni dei driver client sono disponibili anche in questo URL:

[www.filemaker.com/it/support](http://www.filemaker.com/it/support)

Se il file di database FileMaker verrà ospitato usando FileMaker Server Advanced, rendere i driver client disponibili per gli utenti remoti.

Dopo che il driver client necessario è stato installato, è possibile configurarlo per accedere all'origine dati FileMaker e costruire query SQL (Structured Query Language) per interagire con i dati.

Il driver client JDBC è la porzione di driver del software FileMaker che consente alle applicazioni di terze parti o alle applicazioni personalizzate di accedere ai file FileMaker come origini dati JDBC.

### **Requisiti software**

Per installare e utilizzare i driver client JDBC, è necessario JDK 1.4 o successivo.

Per sapere quale versione di Java è in esecuzione sul proprio computer, aprire una finestra di comando (Windows) o la finestra terminale (Mac OS) e digitare `java -version`.

### **Requisiti di rete**

Se si accede ad un'origine dati FileMaker ospitata su un computer diverso, è necessario accedere alla rete mediante il protocollo TCP/IP.

### **Installazione del driver client JDBC**

È necessario avere l'accesso in scrittura alla cartella in cui si installa il driver client JDBC.

#### **Installazione del driver client JDBC:**

1. Eseguire una delle seguenti operazioni:
  - Windows: Se il software è stato scaricato elettronicamente, fare doppio clic sull'icona di installazione (file .exe).
  - Mac OS: Se il software è stato scaricato elettronicamente, fare doppio clic sull'icona con l'immagine del disco (file .dmg).
  - Se si dispone di un disco di installazione, inserirlo nell'apposita unità.
2. Nella finestra FileMaker Pro o Filemaker Server, fare doppio clic sulla cartella xDBC.
3. Nella cartella xDBC fare doppio clic sulla cartella JDBC Client Driver Installer.

4. Copiare il file `fmjdbc.jar` nella cartella adatta al proprio sistema operativo:

- Windows: Copiare il file `fmjdbc.jar` nella cartella che contiene il file eseguibile Java (`java.exe`) o in un altro percorso cartella compreso nel ClassPath dell'applicazione Java.
- Mac OS: Copiare il file `fmjdbc.jar` nella cartella `/Libreria/Java/Extensions` o in un altro percorso cartella compreso nel ClassPath dell'applicazione Java.

Il driver client JDBC è ora disponibile; permette di configurare l'accesso a un'origine dati FileMaker.

## Uso del driver client JDBC

L'applicazione o l'applet Java devono registrare il driver client JDBC nella gestione driver JDBC; inoltre è necessario specificare l'URL esatto nell'applicazione o nell'applet.

**Importante** La porta 2399 deve essere riservata per il driver client JDBC di FileMaker. Il numero di porta sarà sempre 2399. Non è possibile spostare la condivisione JDBC ad una porta diversa.

Per ulteriori informazioni sull'utilizzo del driver client JDBC, vedere capitolo 6, "Uso di JDBC per condividere dati FileMaker."

# Capitolo 6

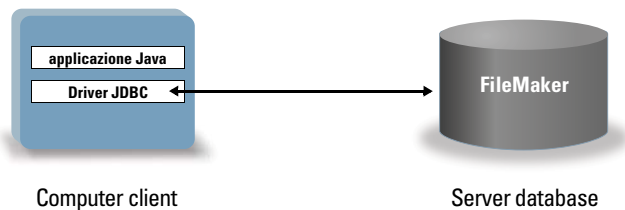
## Uso di JDBC per condividere dati FileMaker

I programmatori Java possono usare il driver client JDBC con qualsiasi strumento Rapid Application Development (RAD) per creare visivamente un'applicazione o un applet Java che si connetta all'origine dati FileMaker. L'applicazione o l'applet Java che usa il driver client JDBC può accedere direttamente ai dati in un file di database FileMaker.

### Informazioni su JDBC

JDBC è un'API Java che esegue istruzioni SQL, il linguaggio standard usato per accedere ai database relazionali. JDBC è un nome e non un acronimo, nonostante si creda comunemente che sia l'acronimo di "Java Database Connectivity" perché è l'equivalente Java di ODBC. JDBC è un'interfaccia di basso livello, cioè viene usata per richiamare direttamente i comandi SQL. Può essere usata anche come base per strumenti e interfacce di livello superiore.

L'applicazione o l'applet Java può interagire direttamente con il file di database FileMaker usando il driver client JDBC. Le istruzioni SQL vengono trasmesse all'host del file di database FileMaker e i risultati di tali istruzioni vengono rispediti al mittente. Se si usa FileMaker Server come host, il file di database FileMaker usato come origine dati può trovarsi su un altro computer (il computer server) connesso alla rete, mentre l'applicazione client o l'applet Java si trovano sul proprio computer (computer client). Questa relazione è detta configurazione client/server.



### Uso del driver client JDBC

È possibile usare il driver client JDBC con un compilatore Java o con uno strumento RAD per effettuare la connessione al database mentre si crea il codice per l'applet o l'applicazione Java. Una volta creato l'applet o l'applicazione Java, affinché questi siano in grado di comunicare con il database è necessario che il driver client JDBC sia presente con i file o compreso nel codice.

Per utilizzare il driver client JDBC, l'applicazione o l'applet Java devono registrare il driver nella gestione driver JDBC; inoltre è necessario specificare l'URL esatto di JDBC nell'applicazione o nell'applet. È necessario l'URL di JDBC per effettuare la connessione al database.

### Driver client JDBC

Il driver client JDBC fornisce supporto parziale per la specifica JDBC 3.0. Le seguenti funzioni non sono supportate da FileMaker:

- Supporto del punto di salvataggio
- Recupero di chiavigenerate automaticamente

- Trasferimento di parametri ad un oggetto istruzione richiamabile per nome
- Supporto cursore aperto
- Recupero e aggiornamento dell'oggetto a cui si fa riferimento mediante un oggetto Ref
- Aggiornamento delle colonne contenenti i tipi di dati CLOB, ARRAY e REF
- tipo di dati booleano
- tipo di dati DATALINK
- Gruppi di trasformazione e la mappatura del tipo
- Relazione tra JDBC SPI e l'architettura Connettore

Per ulteriori informazioni, visitare il sito [www.filemaker.com/it/support](http://www.filemaker.com/it/support).

Il driver client JDBC è stato testato rispetto a Java Development Kit (JDK) 1.5 (Mac OS) e 1.6 (Windows). È un driver Type 4, un driver Java puro con protocollo originale che converte le chiamate JDBC direttamente nel protocollo di rete usato da FileMaker. Questo tipo di driver offre tutti i vantaggi Java, compresa l'installazione automatica (ad esempio, lo scaricamento del driver JDBC con un'applet che lo usa).

La classe del driver e il punto di ingresso principale del driver sono:

`com.filemaker.jdbc.Driver`

**Importante** Il driver client JDBC sostituisce il driver JDBC FileMaker Pro rilasciato con una versione precedente di FileMaker. Se in precedenza è stato configurato l'accesso ad un'origine dati FileMaker usando il vecchio driver, sarà necessario ridefinire l'accesso usando e configurando il nuovo driver.

**Nota** Per verificare se FileMaker xDBC Listener è in esecuzione, è possibile utilizzare il Pannello di controllo (Mac OS) oppure il Task Manager (Windows) per controllare lo stato del processo FileMaker xDBC Listener. Quando questo processo viene avviato, gli viene assegnato il nome `fmjdbc_listener`, utilizzato anche per registrare gli eventi. Il processo FileMaker xDBC Listener è distinto dal processo FileMaker Server.

## Uso di un URL JDBC per collegarsi al database

A molte risorse Java si accede mediante gli URL (Uniform Resource Locators). Gli URL JDBC vengono utilizzati per identificare il database, in modo che il driver client JDBC possa riconoscere e stabilire una connessione con il database.

L'URL JDBC è composto da tre parti principali, separate dal segno di due punti:

`jdbc:<protocollo secondario>:<nome secondario>`

La prima parte nell'URL JDBC è sempre il protocollo JDBC ("`jdbc`"). Il *protocollo secondario* è il nome del driver o il nome del meccanismo che supporta più driver. Per il driver client JDBC, il protocollo secondario è `filemaker`. Il *nome secondario* è l'indirizzo IP della macchina che ospita l'origine dati FileMaker.

## Registrazione del driver client JDBC e connessione ad un'origine dati FileMaker (un esempio)

Di seguito viene illustrato uno snippet di un'applicazione JDBC che:

1. Registra il driver client JDBC nella gestione driver JDBC.
2. Stabilisce una connessione con l'origine dati FileMaker. L'URL JDBC è `jdbc:filemaker://192.168.1.1/database`

**3.** Restituisce codici di errore.

```

import java.sql.*;
class FMPJDBCTest
{
    public static void main(String[ ] args)
    {
        // registra il driver client JDBC
        try {
            Driver d =
            (Driver)Class.forName("com.filemaker.jdbc.Driver").newInstance();
        } catch (Exception e) {
            System.out.println(e);
        }
        // stabilisce una connessione con FileMaker
        Connection con;
        try {
            con = DriverManager.getConnection("jdbc:filemaker://192.168.1.1/mydatabase","username", "password");
        } catch (Exception e) {
            System.out.println(e);
        }
        // avviso connessione
        SQLWarning warning = null;
        try {
            warning = con.getWarnings();
            if (warning == null) {
                System.out.println("No warnings");
                return;
            }
            while (warning != null) {
                System.out.println("Warning: "+warning);
                warning = warning.getNextWarning();
            }
        } catch (Exception e) {
            System.out.println(e);
        }
    }
}
}

```

**Nota** Questo esempio non è stato ideato per essere compilato.

## Specificazione delle proprietà del driver nel nome secondario dell'URL

Specificare le proprietà del driver (nome utente e password) nel nome secondario dell'URL JDBC. Si tratta di proprietà che possono essere passate alla connessione quando si richiama il metodo `DriverManager.getConnection` mediante il parametro `Proprietà`.

- `utente`: un account nel file di database FileMaker che usa un set di privilegi con il privilegio esteso `Accesso tramite ODBC/JDBC`
- `password`: la password per l'account nel file di database FileMaker

### Connessione URL JDBC con il nome del database specificato nell'URL

Formato:

```
jdbc:filemaker://<indirizzo IP host filemaker>/<nomedatabase>
```

Esempio:

```
jdbc:filemaker://192.168.1.1/pubblicazioni
```

### Connessione URL JDBC con il nome del database, il nome utente e la password specificati nell'URL

Formato:

```
jdbc:filemaker://<indirizzo IP host filemaker>/  
<nomedatabase>?user=<nomeutente>&password=<password>
```

Esempio:

```
jdbc:filemaker://192.168.1.1/clienti?user=Riscossioni&password=admin
```

**Nota** Poiché in questa sintassi viene utilizzata la *e commerciale* (&), questo carattere non può essere utilizzato nel nome utente o nella password.

Esempio di nome utente non valido:

```
jdbc:filemaker://hostlocale/vendite_db?user=ad&min&password=admin
```

Esempio di password non valida:

```
jdbc:filemaker://hostlocale/vendite_db?user=admin1&password=ad&min
```

## Soluzioni con più file di database FileMaker

Se la soluzione FileMaker utilizza molti file di database FileMaker, creare un ulteriore file di database che contenga tutti i riferimenti necessari alle origini dati esterne, alle ricorrenze tabella e alle relazioni per la soluzione. Quindi definire questo file di database aggiuntivo come origine dati nell'URL JDBC. Tutti i file di database FileMaker devono essere sullo stesso computer.

## Verifica accesso tramite JDBC

Durante la verifica dell'accesso ad un file di database FileMaker tramite JDBC, assicurarsi che:

- Il file di database FileMaker sia ospitato e disponibile.
- L'account FileMaker utilizzi un set di privilegi con il privilegio esteso di **Accesso tramite ODBC/JDBC**.
- Per l'applicazione host FileMaker Pro o FileMaker Server Advanced sia stata configurata la condivisione tramite ODBC/JDBC.

Per condividere un file di database FileMaker come origine dati, utilizzare FileMaker Pro per definire gli account che devono avere accesso al file di database. Quindi, controllare l'accesso al file di database assegnando set di privilegi agli account, compreso il privilegio esteso di accesso attraverso ODBC/JDBC. Infine, attivare la condivisione dei dati mediante ODBC/JDBC nell'applicazione host FileMaker Server Advanced o FileMaker Pro. Per ulteriori informazioni, vedere la Guida di FileMaker Pro.

- La registrazione del driver client JDBC e l'URL JDBC siano corretti (il driver può essere incluso nell'applicazione Java o situato sul computer client).

Per ulteriori informazioni sull'utilizzo di JDBC per condividere i dati FileMaker, visitare il sito [www.filemaker.com/it/support](http://www.filemaker.com/it/support).



# Capitolo 7

## ***Standard supportati***

In questo capitolo sono descritti le istruzioni e i costrutti SQL supportati dai driver client ODBC e JDBC di FileMaker. Utilizzare i driver client per accedere a una soluzione di database FileMaker da un'applicazione compatibile con ODBC o JDBC. La soluzione di database FileMaker può essere ospitata da FileMaker Pro o FileMaker Server Advanced.

Il driver client ODBC supporta ODBC 3.5 livello 1 con alcune funzioni del livello 2. Il driver client JDBC fornisce supporto parziale per la specifica JDBC 3.0. Per ulteriori informazioni, vedere [www.filemaker.com/it/support](http://www.filemaker.com/it/support). I driver client ODBC e JDBC supportano la conformità SQL-92 entry-level, con alcune funzioni intermedie SQL-92.

### **Supporto caratteri Unicode**

I driver client ODBC e JDBC supportano le API Unicode. Tuttavia, se si sta creando un'applicazione personalizzata che usa i driver client, usare ASCII per i nomi dei campi, i nomi delle tabelle e i nomi dei file (in caso venissero utilizzati strumenti o applicazioni di query diversi da Unicode).

**Nota** Per inserire e recuperare i dati Unicode, utilizzare SQL\_C\_WCHAR.

### **Istruzioni SQL**

I driver client ODBC e JDBC supportano le seguenti istruzioni SQL:

SELECT (vedere di seguito)      DELETE (pagina 44)      INSERT (pagina 44)      UPDATE (pagina 45)  
CREATE TABLE (pagina 46)      ALTER TABLE (pagina 47)      CREATE INDEX (pagina 47)      DROP INDEX (pagina 48)

I driver client supportano anche la mappatura dei dati di tipo FileMaker su dati di tipo ODBC SQL e JDBC SQL. Per informazioni sulle conversioni dei tipi di dati vedere “Mappatura di campi FileMaker su dati di tipo ODBC” a pagina 61 e “Mappatura di campi FileMaker su dati di tipo JDBC” a pagina 62. Per ulteriori informazioni sulla creazione di query SQL, consultare un manuale di terze parti.

**Nota** I driver client ODBC e JDBC non supportano i portali FileMaker.

### **Istruzione SELECT**

Utilizzare l'istruzione SELECT per specificare le colonne richieste. Far seguire l'istruzione SELECT dalle espressioni di colonna (simili ai nomi di campo) che si desidera recuperare, ad esempio `cognome`). Le espressioni possono includere operazioni matematiche o manipolazioni di stringhe, ad esempio `STIPENDIO * 1.05`.

L'istruzione SELECT può utilizzare varie clausole:

```
SELECT [DISTINCT] { * | espressione_colonna [[AS] alias_colonna], ... }
FROM nome_tabella [alias_tabella], ...
[ WHERE espr1 operatore_rel espr2 ]
[ GROUP BY {espressione_colonna, ...} ]
[ HAVING espr1 operatore_rel espr2 ]
[ UNION [ALL] (SELECT..) ]
[ ORDER BY {espressione_ordinamento [DESC | ASC]}, ... ]
[ FOR UPDATE [OF {espressione_colonna, ...}] ]
```

Gli elementi racchiusi tra parentesi sono facoltativi.

alias\_colonna può essere utilizzato per assegnare alla colonna un nome più descrittivo, o per abbreviare il nome di una colonna più lunga. Ad esempio per assegnare l'alias settore alla colonna sett:

```
SELECT sett AS settore FROM dip
```

Davanti ai nomi dei campi possono essere aggiunti il nome della tabella o l'alias della tabella. Ad esempio, EMP.LAST\_NAME o ECOGNOME, dove E è l'alias della tabella EMP (dip).

L'operatore DISTINCT può precedere la prima espressione di colonna. Questo operatore elimina le righe doppie dal risultato di una query. Ad esempio:

```
SELECT DISTINCT sett FROM dip
```

## clausole SQL

I driver client ODBC e JDBC supportano le seguenti clausole SQL.

Utilizzare questa clausola SQL	Per
FROM (pagina 39)	Indicare le tabelle utilizzate nell'istruzione SELECT.
WHERE (pagina 39)	Specificare le condizioni che i record devono soddisfare per essere recuperati, come nel caso di una richiesta di ricerca di FileMaker Pro.
GROUP BY (pagina 40)	Specificare i nomi di uno o più campi in base a cui raggruppare i valori restituiti. Questa clausola viene utilizzata per restituire una serie di valori aggregati tramite la restituzione di una riga per ciascun gruppo, come nel caso di un riassunto parziale di FileMaker Pro.
HAVING (pagina 40)	Specificare le condizioni per i gruppi di record, ad esempio per visualizzare solo i settori per cui l'importo complessivo degli stipendi è superiore a 200.000 Euro.
UNION (pagina 39)	Combinare i risultati di due o più istruzioni SELECT.
ORDER BY (pagina 41)	Specificare l'ordinamento dei record.
FOR UPDATE (pagina 41)	Per eseguire gli aggiornamenti o le eliminazioni nella posizione attraverso i cursori SQL

**Nota** Se si tenta di recuperare i dati da una tabella senza colonne, l'istruzione SELECT non restituisce nulla.

## Clausola FROM

La clausola FROM indica le tabelle che vengono utilizzate nell'istruzione SELECT. Il formato è:

```
FROM [, nome_tabella [alias_tabella] [, nome_tabella [alias_tabella]
nome_tabella è il nome di una tabella nel database corrente.
```

alias\_tabella può essere utilizzato per assegnare alla tabella un nome più descrittivo, per abbreviare un nome di tabella lungo o per includere la stessa tabella nella query più di una volta (ad esempio, in auto-collegamenti).

Davanti ai nomi dei campi possono essere aggiunti il nome della tabella o l'alias della tabella. Ad esempio, nel caso di una tabella FROM dipendente E, è possibile indicare il campo COGNOME come E COGNOME. Gli alias di tabella devono essere utilizzati se l'istruzione SELECT unisce una tabella a se stessa. Ad esempio:

```
SELECT * FROM dipendente E, dipendente F WHERE E.id_manager =
F.id_dipendente
```

Il segno uguale (=) comprende soltanto le righe corrispondenti nei risultati.

Per unire più di una tabella ed eliminare tutte le righe che non hanno righe corrispondenti in entrambe le tabelle di origine, è possibile usare INNER JOIN. Ad esempio:

```
SELECT*
FROM Venditori INNER JOIN Dati_Vendite
ON Venditori.ID_Venditore = Dati_Vendite.ID_Venditore
```

Per unire due tabelle senza eliminare righe nella prima tabella (quella "a sinistra"), è possibile utilizzare LEFT JOIN.

```
SELECT*
FROM Venditori LEFT JOIN Dati_Vendite
ON Venditori.Venditore ID = Dati_Vendite.IDSettore
```

Ogni riga della tabella "Venditori" appare nella tabella unita.

Per unire due tabelle senza eliminare righe nella seconda tabella (quella "a destra"), è possibile utilizzare RIGHT JOIN.

```
SELECT*
FROM Venditori RIGHT JOIN Dati_Vendite
ON Venditori.Venditore ID = Dati_Vendite.IDSettore
```

Ogni riga della tabella "Dati vendite" appare nella tabella unita.

### Note

- LEFT JOIN è supportato, ma la grammatica LEFT OUTER JOIN attualmente non è supportata.
- RIGHT JOIN è supportato, ma la grammatica RIGHT OUTER JOIN attualmente non è supportata.
- FULL OUTER JOIN non è supportato attualmente.

## Clausola WHERE

La clausola WHERE specifica le condizioni che i record devono soddisfare per essere recuperati.

La clausola WHERE contiene le condizioni nella forma:

```
WHERE espr1 operatore_rel espr2
```

espr1 e espr2 possono essere nomi di campo, valori costanti o espressioni.

operatore\_rel è l'operatore relazionale che collega le due espressioni. Ad esempio, la seguente istruzione SELECT recupera i nomi dei dipendenti con stipendio uguale o superiore a 20.000 Euro.

```
SELECT cognome,nome FROM dip WHERE stipendio >= 20000
```

La clausola WHERE può utilizzare anche espressioni come:

```
WHERE expr1 IS NULL
WHERE NOT espr2
```

**Nota** Se si utilizzano nomi completamente qualificati nella lista (proiezione) SELECT, è necessario utilizzare anche nomi completamente qualificati nella clausola WHERE correlata.

## Clausola GROUP BY

La clausola GROUP BY specifica i nomi di uno o più campi in base a cui raggruppare i valori restituiti. Questa clausola viene utilizzata per restituire un gruppo di valori aggregati. Ha il seguente formato:

```
GROUP BY colonne
```

colonne deve corrispondere all'espressione della colonna utilizzata nella clausola SELECT.

Un'espressione di colonna può essere costituita da uno o più nomi di campi della tabella di database separati da virgole.

L'esempio seguente somma gli stipendi di ogni settore.

```
SELECT id_sett, SUM(stipendio) FROM dip GROUP BY id_sett
```

Questa istruzione restituisce una riga per ogni singolo ID settore. Ogni riga contiene l'ID settore e la somma degli stipendi dei dipendenti del settore.

## Clausola HAVING

La clausola HAVING permette di specificare le condizioni per i gruppi di record, ad esempio per visualizzare solo i settori per cui l'importo complessivo degli stipendi è superiore a 200.000 Euro. Ha il seguente formato:

```
HAVING espr1 operatore_rel espr2
```

espr1 e espr2 possono essere nomi di campo, valori costanti o espressioni. Queste espressioni non devono corrispondere all'espressione di una colonna nella clausola SELECT.

operatore\_rel è l'operatore relazionale che collega le due espressioni. Nell'esempio che segue vengono restituiti solo i settori per cui l'importo complessivo degli stipendi è superiore a 200.000 Euro:

```
SELECT id_sett, SUM (stipendio) FROM dip
GROUP BY id_sett HAVING SUM (stipendio) > 200000
```

## Operatore UNION

L'operatore UNION combina i risultati di due o più istruzioni SELECT in un solo risultato. Il singolo risultato ottenuto comprende tutti i record restituiti dalle istruzioni SELECT. Per impostazione predefinita, i record duplicati non vengono restituiti. Per restituire i record duplicati, utilizzare la parola chiave All (UNION ALL). Il formato è:

```
SELECT istruzione UNION [ALL] SELECT istruzione
```

Quando si usa l'operatore UNION, le liste di selezione di ogni istruzione SELECT devono avere lo stesso numero di espressioni di colonna, con dati dello stesso tipo, e devono essere specificate nello stesso ordine. Ad esempio:

```
SELECT cognome, stipendio, data_assunzione FROM dip UNION SELECT nome,
pagamento, data_nascita FROM persona
```

In questo esempio vi è lo stesso numero di espressioni di colonna e ogni espressione di colonna, in ordine, ha dati dello stesso tipo.

L'esempio che segue non è valido perché i tipi di dati delle espressioni di colonna sono diversi (STIPENDIO from IMP ha dati di tipo diverso rispetto a COGNOME from AUMENTI). In questo esempio vi è lo stesso numero di espressioni di colonna in ogni istruzione SELECT, ma le espressioni non sono nello stesso ordine per tipo di dati.

```
SELECT cognome, stipendio FROM dip UNION SELECT stipendio, cognome FROM
aumenti
```

## Clausola ORDER BY

La clausola ORDER BY indica il modo in cui i record devono essere ordinati. Il formato è:

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

espressione\_ordinamento può essere: nomi di campo, espressioni o il numero di posizione dell'espressione di colonna da usare. Per impostazione predefinita l'ordinamento viene effettuato in modo crescente (ASC).

Ad esempio, per ordinare in base al cognome e poi al nome, è possibile usare una delle seguenti istruzioni SELECT:

```
SELECT id_dip, cognome, nome FROM dip ORDER BY cognome, nome
```

oppure

```
SELECT id_dip, cognome, nome FROM dip ORDER BY 2,3
```

Nel secondo esempio, cognome è la seconda espressione di colonna dopo SELECT, pertanto ORDER BY 2 ordina in base al cognome.

## Clausola FOR UPDATE

La clausola FOR UPDATE blocca gli aggiornamenti o le eliminazioni nella posizione attraverso i cursori SQL. Il formato è:

```
FOR UPDATE [OF espressioni_colonna]
```

espressioni\_colonna è una lista di nomi di campi nella tabella di database che si desidera aggiornare, separati da una virgola. espressioni\_colonna è opzionale e viene ignorato.

L'esempio che segue restituisce tutti i record nel database dei dipendenti per cui il valore del campo STIPENDIO è superiore a 20.000 Euro. I record recuperati vengono bloccati. Se il record viene aggiornato o eliminato, il blocco viene mantenuto finché non si applica la modifica. In caso contrario, il blocco viene rilasciato quando si recupera il record successivo.

```
SELECT * FROM dip WHERE stipendio > 20000
FOR UPDATE OF cognome, nome, stipendio
```

**Altri esempi:**

Usando	Esempio SQL
costante di testo	<code>SELECT 'CatDog' FROM Venditori</code>
costante numerica	<code>SELECT 999 FROM Venditori</code>
costante di data	<code>SELECT DATE '05.06.11' FROM Venditori</code>
costante di ora	<code>SELECT TIME '02:49:03' FROM Venditori</code>
costante Indicatore data e ora	<code>SELECT TIMESTAMP '05.06.11 02:49:03' FROM Venditori</code>
colonna di testo	<code>SELECT Nome_Azienda FROM Dati_Vendite</code> <code>SELECT DISTINCT NomeAzienda FROM Dati_Vendite</code>
colonna numerica	<code>SELECT Quantità FROM Dati_Vendite</code> <code>SELECT DISTINCT Quantità FROM Dati_Vendite</code>
colonna data	<code>SELECT Data_Vendita FROM Dati_Vendite</code> <code>SELECT DISTINCT Data_Vendita FROM Dati_Vendite</code>
colonna ora	<code>SELECT Ora_Vendita FROM Dati_Vendite</code> <code>SELECT DISTINCT Ora_Vendita FROM Dati_Vendite</code>
colonna Indicatore data e ora	<code>SELECT IndicatoreDataOra_Vendita FROM Dati_Vendite</code> <code>SELECT DISTINCT IndicatoreDataOra_Vendita FROM Dati_Vendite</code>
Colonna <sup>a</sup> BLOB	<code>SELECT Brochure_Società FROM Dati_Vendite</code> <code>SELECT RicavaCome(Logo_Società, 'JPEG') FROM Dati_Vendite</code>
carattere jolly*	<code>SELECT * FROM Venditori</code> <code>SELECT DISTINCT * FROM Venditori</code>

a. Un BLOB è un campo Contenitore di un file di database FileMaker.

**Note sugli esempi**

Una colonna è un riferimento ad un campo nel file di database FileMaker (il campo può contenere più valori distinti).

Il carattere jolly asterisco (\*) rappresenta "tutto". Per l'esempio `SELECT * FROM Venditori`, il risultato comprende tutte le colonne nella tabella `Venditori`. Per l'esempio `SELECT DISTINCT * FROM Venditori`, il risultato è costituito da tutte le righe uniche nella tabella `Venditori` (senza duplicati).

- FileMaker non memorizza dati per stringhe vuote, quindi le seguenti query non restituiscono mai record:

```
SELECT * FROM test WHERE c ="
```

```
SELECT * FROM test WHERE c <>"
```

- Se si usa `SELECT` con dati binari, è necessario utilizzare la funzione `RicavaCome()` per specificare il flusso da restituire. Per ulteriori informazioni, vedere la seguente sezione “Recupero dei contenuti di un campo Contenitore: Funzione `CAST()` e funzione `RicavaCome()`”.

**Recupero dei contenuti di un campo Contenitore: Funzione `CAST()` e funzione `RicavaCome()`**

Da un campo Contenitore si possono recuperare dati binari, informazioni di riferimento al file o dati di un tipo specifico di file.

Per recuperare i dati binari utilizzare un'istruzione `SELECT` standard. Ad esempio:

```
SELECT Brochure_Società FROM Dati_Vendite
```

Se esistono file o dati JPEG, l'istruzione `SELECT` recupera i dati in forma binaria; in caso contrario, l'istruzione `SELECT` restituisce `<null>`.

Per recuperare le informazioni di riferimento al file da un campo Contenitore (come il percorso di un file, di un'immagine o di un filmato Quicktime) utilizzare la funzione CAST con un'istruzione SELECT.

Ad esempio:

```
SELECT CAST(Brochure_Società AS VARCHAR(NNN)) FROM Dati_Vendite
```

In questo esempio se:

- È stato inserito un file nel campo Contenitore utilizzando FileMaker Pro, ma è stato memorizzato solo un riferimento al file, l'istruzione SELECT recupera le informazioni di riferimento al file come SQL\_VARCHAR.
- Sono stati inseriti i contenuti di un file nel campo Contenitore utilizzando FileMaker Pro, l'istruzione SELECT recupera il nome del file.
- È stato importato un file nel campo Contenitore da un'altra applicazione, l'istruzione SELECT visualizza '?' (Il file viene visualizzato come Senza nome.dat in FileMaker Pro).

Per recuperare i dati di un campo Contenitore, utilizzare la funzione RicavaCome e specificare il tipo di file a seconda del modo in cui i dati sono stati inseriti nel campo Contenitore in FileMaker Pro.

- Se i dati sono stati inseriti utilizzando il comando **Inserisci > File**, specificare 'FILE' nella funzione RicavaCome. Ad esempio:

```
SELECT RicavaCome(Brochure_Società, 'FILE') FROM Dati_Vendite
```

- Se i dati sono stati inseriti utilizzando il comando **Inserisci > Suono** (Suono standard — formato grezzo Mac OS), specificare 'snd' nella funzione RicavaCome. Ad esempio:

```
SELECT RicavaCome(Meeting_Società_, 'snd') FROM Newsletter_Società
```

- Se i dati sono stati inseriti utilizzando il comando **Inserisci > Oggetto** (Dati Contenitore OLE), specificare 'EMBO' nella funzione RicavaCome. Ad esempio:

```
SELECT RicavaCome(Risultati_Società, 'EMBO') FROM Resoconto_Annuale
```

- Se i dati sono stati inseriti utilizzando il comando **Inserisci > Immagine**, trascinandoli oppure incollandoli dagli appunti, specificare uno dei tipi di file elencati nella seguente tabella. Ad esempio:

```
SELECT RicavaCome(Logo_Società, 'JPEG') FROM Icone_Società
```

Tipo di file	Descrizione	Tipo di file	Descrizione
'EMF+'	Windows Enhanced Metafile Plus	'PDF '	Portable Document Format
'EPS '	PostScript incorporato	'PICT'	Mac OS (non ha intestazioni basate su file di 512 byte)
'FPix'	Flash (FPX)	'PNGf'	Formato immagine Bitmap
'FORK'	Fork di risorse (Mac OS)	'PNTG'	MacPaint
'GIF'	Graphics Interchange Format	'qtif'	File immagine QuickTime
'JPEG'	Immagini fotografiche	'SGI'	Formato generico bitmap
'JP2'	JPEG 2000	'TIFF'	Formato raster del file per immagini digitali
'META'	Windows Metafile (avanzato)	'TPIC'	Targa
'METO'	Windows Metafile (originale)	'XMLO'	Oggetti del formato.
'moov'	Vecchio formato QuickTime (Mac OS)	'8BPS'	Photoshop (PSD)

## Istruzione DELETE

Usare l'istruzione DELETE per eliminare i record da una tabella di database. Il formato dell'istruzione DELETE è:

```
DELETE FROM nome_tabella [ WHERE { condizioni } ]
```

**Nota** La clausola WHERE determina i record da eliminare. Se non si include la parola chiave WHERE, tutti i record nella tabella vengono cancellati (ma la tabella rimane invariata).

Di seguito è riportato un esempio di istruzione DELETE eseguita sulla tabella Dipendente:

```
DELETE FROM dip WHERE id_dip = 'E10001'
```

L'istruzione DELETE rimuove tutti i record che soddisfano le condizioni della clausola WHERE. In questo caso vengono eliminati tutti i record in cui il codice del dipendente è E10001. Poiché nella tabella Dipendenti i codici dei dipendenti sono unici, viene eliminato un solo record.

## Istruzione INSERT

Utilizzare l'istruzione INSERT per creare record in una tabella di database. È possibile specificare:

- Una lista di valori da inserire come nuovo record
- Un'istruzione SELECT che copia i dati di un'altra tabella da inserire come gruppo di nuovi record

Il formato dell'istruzione INSERT è:

```
INSERT INTO nome_tabella [(nome_colonna, ..)] VALUES (espr, ..).  
[, VALUES (expr,..)]
```

nome\_colonna è una lista facoltativa di nomi di colonna che specifica il nome e l'ordine delle colonne di cui sono stati specificati i valori nella clausola VALUES. Se si omette nome\_colonna, le espressioni di valore (espr) dovranno specificare i valori di tutte le colonne definite nella tabella e dovranno riflettere l'ordine delle colonne definito per la tabella. nome\_colonna può anche specificare la ripetizione di un campo, per esempio lastDates [4].

espr è la lista di espressioni che forniscono i valori delle colonne del nuovo record. In genere, le espressioni sono valori costanti per le colonne (ma possono anche essere una subquery). È necessario racchiudere i valori delle stringhe di caratteri tra coppie di virgolette singole ('). Per includere una virgoletta singola nel valore di una stringa di caratteri racchiusa tra virgolette singole, usare due virgolette singole insieme (ad esempio, 'L"aquilone').

Le subquery devono essere racchiuse tra parentesi.

L'esempio che segue inserisce una lista di espressioni:

```
INSERT INTO dip (cognome, nome, id_dip, stipendio, data_assunzione)  
VALUES ('Smith', 'John', 'E22345', 27500, {d '05/06/2008'})
```

Ciascuna istruzione INSERT aggiunge un record nella tabella di database. In questo caso è stato aggiunto un record alla tabella di database dei dipendenti denominata DIP. I valori sono stati specificati per cinque colonne. Alle restanti colonne della tabella è stato assegnato un valore vuoto, ossia Null.

**Nota** Nei campi Contenitore, è possibile inserire (INSERT) solo testo, a meno che si prepari un'istruzione parametrizzata e si effettui lo streaming dei dati dall'applicazione. Per utilizzare dati binari è necessario specificare il tipo nella funzione PutAs(): PutAs(col, 'tipo'), dove il valore del tipo è uno di quelli descritti in “Recupero dei contenuti di un campo Contenitore: Funzione CAST() e funzione RicavaCome()” a pagina 42.

L'istruzione `SELECT` è una query che restituisce valori per ciascun valore `nome_colonna` specificato nella lista dei nomi di colonna. Se si specifica un'istruzione `SELECT` anziché una lista di espressioni di valori, sarà possibile selezionare un gruppo di righe da una tabella e inserirlo in un'altra tabella tramite una singola istruzione `INSERT`.

Di seguito è riportato un esempio di istruzione `INSERT` che utilizza un'istruzione `SELECT`:

```
INSERT INTO dip1 (nome, cognome, id_dip, settore, stipendio)
  SELECT nome, cognome, id_dip, settore, stipendio from dip
  WHERE settore = 'D050'
```

In questo tipo di istruzione `INSERT`, il numero di colonne da inserire deve corrispondere al numero di colonne dell'istruzione `SELECT`. La lista di colonne da inserire deve corrispondere alle colonne nell'istruzione `SELECT`, analogamente a quanto accade per le espressioni di valore nell'altro tipo di istruzione `INSERT`. Ad esempio, la prima colonna inserita corrisponde alla prima colonna selezionata; la seconda alla seconda, e così via.

Le dimensioni e il tipo di dati di queste colonne corrispondenti devono essere compatibili. Ciascuna colonna della lista `SELECT` dovrebbe disporre di un tipo di dati accettato dal driver client ODBC o JDBC per un'istruzione `INSERT/UPDATE` standard della colonna corrispondente nella lista `INSERT`. Se la dimensione dei valori nella colonna della lista `SELECT` supera la dimensione dei valori nella colonna della lista `INSERT` corrispondente, i valori vengono troncati.

L'istruzione `SELECT` viene valutata prima dell'inserimento di qualsiasi valore.

## Istruzione UPDATE

Utilizzare l'istruzione `UPDATE` per cambiare i record in una tabella di database. Il formato dell'istruzione `UPDATE` è:

```
UPDATE nome_tabella SET nome_colonna = espr, ... [ WHERE { condizioni } ]
```

`nome_colonna` è il nome di una colonna di cui si desidera modificare il valore. È possibile modificare più colonne in una singola istruzione.

`espr` è il nuovo valore della colonna.

In genere, le espressioni sono valori costanti per le colonne (ma possono anche essere una subquery). È necessario racchiudere i valori delle stringhe di caratteri tra coppie di virgolette singole (`'`). Per includere una virgoletta singola nel valore di una stringa di caratteri racchiusa tra virgolette singole, usare due virgolette singole insieme (ad esempio, `'L"aquilone'`).

Le subquery devono essere racchiuse tra parentesi.

La clausola `WHERE` può essere una qualsiasi clausola valida. Determina i record da aggiornare.

Di seguito è riportato un esempio di istruzione `UPDATE` eseguita sulla tabella `Dip`:

```
UPDATE dip SET stipendio=32000; detrazioni=1 WHERE id_dip = 'E10001'
```

L'istruzione `UPDATE` modifica tutti i record che soddisfano le condizioni della clausola `WHERE`. In questo caso vengono modificati lo stipendio e lo stato delle detrazioni per tutti i dipendenti il cui codice è `E10001`. Poiché nella tabella `Dipendenti` i codici dei dipendenti sono unici, viene aggiornato un solo record.

Di seguito è riportato un esempio con una subquery:

```
UPDATE dip SET stipendio = (SELECT avg(stipendio) from dip) WHERE id_dip =
'E10001'
```

In questo caso lo stipendio del dipendente il cui codice è E10001 viene sostituito con lo stipendio medio della società.

**Nota** Nei campi Contenitore, è possibile aggiornare (UPDATE) solo con testo, a meno che si prepari un'istruzione parametrizzata e si effettui lo streaming dei dati dall'applicazione. Per utilizzare dati binari è necessario specificare il tipo nella funzione PutAs(): PutAs(col, 'tipo'), dove il valore del tipo è uno di quelli descritti in "Recupero dei contenuti di un campo Contenitore: Funzione CAST() e funzione RicavaCome()" a pagina 42.

## Istruzione CREATE TABLE

Utilizzare l'istruzione CREATE TABLE per creare una tabella in un file di database. Il formato dell'istruzione CREATE TABLE:

```
CREATE TABLE nome_tabella lista_elementi_tabella [DEFAULT expr] [UNIQUE] [NOT NULL]
```

All'interno dell'istruzione, si specifica il nome e il tipo di dati di ogni colonna.

- nome\_tabella e lista\_elementi\_tabella hanno un limite di 100 caratteri.
- La parola chiave DEFAULT permette di impostare un valore predefinito per una colonna. È possibile utilizzare un letterale come CURRENT\_USER, CURRENT\_DATE, CURRENT\_TIME o CURRENT\_TIMESTAMP.
- Se una colonna viene definita UNIQUE si seleziona automaticamente l'opzione di verifica Unique per il campo corrispondente nel file di database FileMaker.
- Se una colonna viene definita NOT NULL, si seleziona automaticamente l'opzione di verifica Non vuoto per il campo corrispondente nel file di database FileMaker. Il campo viene contrassegnato come Valore richiesto nella scheda Campi della finestra di dialogo Definisci database in FileMaker Pro

## Esempi

Usando	Esempio SQL
colonna di testo	CREATE TABLE T1 (C1 VARCHAR, C2 VARCHAR (50), C3 VARCHAR (1001), C4 VARCHAR (500276))
colonna di testo, NOT NULL	CREATE TABLE T1NN (C1 VARCHAR NOT NULL, C2 VARCHAR (50) NOT NULL, C3 VARCHAR (1001) NOT NULL, C4 VARCHAR (500276) NOT NULL)
colonna numerica	CREATE TABLE T2 (C1 DECIMAL, C2 DECIMAL (10,0), C3 DECIMAL (7539,2), C4 DECIMAL (497925,301))
colonna data	CREATE TABLE T3 (C1 DATE, C2 DATE, C3 DATE, C4 DATE)
colonna ora	CREATE TABLE T4 (C1 TIME, C2 TIME, C3 TIME, C4 TIME)
colonna Indicatore data e ora	CREATE TABLE T5 (C1 TIMESTAMP, C2 TIMESTAMP, C3 TIMESTAMP, C4 TIMESTAMP)
Colonna BLOB	CREATE TABLE T6 (C1 BLOB, C2 BLOB, C3 BLOB, C4 BLOB)

## Istruzione ALTER TABLE

Utilizzare l'istruzione ALTER TABLE per cambiare la struttura di una tabella esistente in un file di database. È possibile modificare una sola colonna in ogni istruzione. I formati dell'istruzione ALTER TABLE sono:

```
ALTER TABLE nome_tabella ADD [COLUMN] definizione_colonna
ALTER TABLE nome_tabella DROP [COLUMN] nome_colonna_non_qualificato
ALTER TABLE nome_tabella ALTER [COLUMN] definizione_colonna SET DEFAULT expr
ALTER TABLE nome_tabella ALTER [COLUMN] definizione_colonna DROP DEFAULT
```

È necessario conoscere la struttura della tabella e sapere come modificarla prima di usare l'istruzione ALTER TABLE.

### Esempi

Per	Esempio SQL
aggiungere colonne	ALTER TABLE Venditori ADD (C1 VARCHAR)
rimuovere colonne	ALTER TABLE Venditori DROP C1
impostare il valore predefinito per una colonna	ALTER TABLE Venditori ALTER Società SET DEFAULT 'FileMaker'
rimuovere il valore predefinito per una colonna	ALTER TABLE Venditori ALTER Società DROP DEFAULT

**Nota** SET DEFAULT e DROP DEFAULT non influiscono sulle righe esistenti nella tabella, ma cambiano il valore predefinito per le righe aggiunte successivamente alla tabella.

## Istruzione CREATE INDEX

Utilizzare l'istruzione CREATE INDEX per velocizzare le ricerche nel file di database. Il formato dell'istruzione CREATE INDEX è:

```
CREATE INDEX ON nome_tabella.nome_colonna
CREATE INDEX ON nome_tabella (nome_colonna)
```

CREATE INDEX è supportato per una sola colonna (gli indici a più colonne non sono supportati). Gli indici non sono consentiti sulle colonne che corrispondono a campi di tipo Contenitore, campi Riassunto, campi per cui è prevista l'opzione di memorizzazione globale o campi Calcolo non memorizzati in un file di database FileMaker.

Con la creazione di un indice per una colonna di testo si imposta automaticamente l'opzione di memorizzazione Indicizzazione su Minimo per il campo corrispondente nel file di database FileMaker. Con la creazione di un indice per una colonna non di testo (o per una colonna formattata come testo giapponese) si imposta automaticamente l'opzione di memorizzazione Indicizzazione su Tutti per il campo corrispondente nel file di database FileMaker.

Con la creazione di un indice per tutte le colonne si imposta automaticamente l'opzione di memorizzazione Indicizzazione su Crea automaticamente gli indici quando necessario per il campo corrispondente nel file di database FileMaker.

FileMaker crea automaticamente gli indici secondo necessità. Utilizzando CREATE INDEX l'indice da creare viene generato subito anziché su richiesta.

### Esempio

```
CREATE INDEX ON Venditori.ID_Venditore
```

## Istruzione DROP INDEX

Utilizzare l'istruzione `DROP INDEX` per rimuovere un indice da un file di database. Il formato dell'istruzione `DROP INDEX` è:

```
DROP INDEX ON nome_tabella.nome_colonna
DROP INDEX ON nome_tabella (nome_colonna)
```

Se il file di database è troppo grande o non si usa spesso un campo nelle query, rimuovere l'indice.

Se le query non sono soddisfacenti e si sta lavorando con un file di database FileMaker particolarmente grande, con molti campi di testo indicizzati, è possibile eliminare gli indici da alcuni campi. È anche possibile eliminare gli indici dai campi che si usano raramente nell'istruzione `SELECT`.

Eliminando un indice da qualsiasi colonna, si seleziona automaticamente l'opzione di memorizzazione Nessuno e si elimina automaticamente gli indici quando necessario in Indicizzazione per il campo corrispondente nel file di database FileMaker.

L'attributo `PREVENT INDEX CREATION` non è supportato.

### Esempio

```
DROP INDEX ON Venditori.ID_Venditore
```

## Funzioni aggregate SQL

Le funzioni aggregate restituiscono un solo valore da un gruppo di record. È possibile usare una funzione aggregata come parte dell'istruzione `SELECT`, con il nome di un campo (ad esempio, `AVG(STIPENDIO)`), o in combinazione con un'espressione di colonna (ad esempio, `AVG(STIPENDIO * 1.07)`).

È possibile far precedere all'espressione di colonna l'operatore `DISTINCT` per eliminare i valori duplicati. Ad esempio:

```
COUNT (DISTINCT cognome)
```

In questo esempio vengono contati solo i valori univoci di cognome.

### Esempi

Funzione aggregata	Restituisce
SUM	Il totale dei valori di un'espressione di un campo numerico. Ad esempio, <code>SUM(STIPENDIO)</code> restituisce la somma di tutti i valori del campo stipendio.
AVG	La media dei valori di un'espressione di un campo numerico. Ad esempio, <code>AVG(STIPENDIO)</code> restituisce la media di tutti i valori del campo stipendio.
COUNT	Il numero di valori in qualsiasi espressione campo. Ad esempio, <code>COUNT (NOME)</code> restituisce il numero di valori dei nomi. Quando si usa <code>COUNT</code> con il nome di un campo, <code>COUNT</code> restituisce il numero di valori di campi non "null". Un esempio particolare è <code>COUNT (*)</code> , che restituisce il numero di record del gruppo, compresi i record con valori nulli.
MAX	Il valore massimo in qualsiasi espressione campo. Ad esempio, <code>MAX(STIPENDIO)</code> restituisce il valore massimo del campo stipendio.
MIN	Il valore minimo in qualsiasi espressione campo. Ad esempio, <code>MIN(STIPENDIO)</code> restituisce il valore minimo del campo stipendio.

```
SELECT SUM (Dati_Vendite.Quantità) AS agg FROM Dati_Vendite
SELECT AVG (Dati_Vendite.Quantità) AS agg FROM Dati_Vendite
SELECT COUNT (Dati_Vendite.Quantità) AS agg FROM Dati_Vendite
```

```

SELECT MAX (Dati_Vendite.Quantità) AS agg FROM Dati_Vendite
WHERE Dati_Vendite.Quantità < 3000
SELECT MIN (Dati_Vendite.Quantità) AS agg FROM Dati_Vendite
WHERE Dati_Vendite.Quantità > 3000

```

## Espressioni SQL

Utilizzare le espressioni nelle clausole WHERE, HAVING e ORDER BY delle istruzioni SELECT per generare query dettagliate e sofisticate. Elementi validi sono:

Nomi di campo	Operatori numerici	Operatori relazionali
Costanti	Operatori alfabetici	Operatori logici
Notazione esponenziale	Operatori data	Funzioni

### Nomi di campo

L'espressione più comune è un semplice nome di campo, come ad esempio `calc o Dati_vendita.Fattura_ID`.

### Costanti

Le costanti sono valori che non cambiano. Ad esempio, nell'espressione `PREZZO * 1.05`, il valore 1.05 è una costante. In alternativa è possibile assegnare il valore 30 alla costante `Numero_Di_Giorni_A_Giugno`. È necessario racchiudere le costanti di caratteri tra coppie di virgolette singole ('). Per includere una virgoletta singola in una costante di caratteri racchiusa tra virgolette singole, usare due virgolette singole insieme (ad esempio, 'L"aquilone').

FileMaker accetta le costanti di data, ora e indicatore data e ora in formato ODBC/JDBC in parentesi graffe ({}), ad esempio:

- {D '2010-06-05'}
- {T '14:35:10'}
- {TS '2010-06-05 14:35:10'}

FileMaker accetta anche la sintassi SQL-92 con data e ora in formato ISO, senza parentesi:

- DATA 'AAAA-MM-GG'
- ORA 'HH:MM:SS'
- IINDICATORE DATA E ORA 'AAAA-MM-GG HH:MM:SS'

Costante	Sintassi accettabile (esempi)
Testo	'Parigi'
Numero	1.05
Data	DATE '05.06.2010' { D '05.06.2010' {06/05/2010} {06/05/10}

**Nota:** La sintassi dell'anno a 2 cifre non è supportata per il formato ODBC/JDBC o per il formato SQL-92.

Costante	Sintassi accettabile (esempi)
Ora	TIME '14:35:10' { T '14:35:10' } { 14:35:10 }
Indicatore data e ora	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 } Verificare che Tipo di dati restrittivo: Data a 4 cifre non sia selezionato come opzione di verifica nel file di database FileMaker per un campo che utilizza questa sintassi dell'anno a 2 cifre. <b>Nota:</b> La sintassi dell'anno a 2 cifre non è supportata per il formato ODBC/JDBC o per il formato SQL-92.

Quando si inseriscono valori di data e ora utilizzare il formato locale del file di database. Ad esempio, se il database è stato creato in un sistema di lingua italiana, utilizzare i formati data e ora italiani.

## Notazione esponenziale/scientifica

I numeri possono essere espressi utilizzando la notazione scientifica.

### Esempio

```
SELECT colonna1 / 3.4E+7 FROM tabella1 WHERE calc < 3.4E-6 * colonna2
```

## Operatori numerici

È possibile comprendere i seguenti operatori nelle espressioni numeriche: +, -, \*, /, and ^ o \*\* (esponenziazione).

È possibile anteporre alle espressioni numeriche un più (+) oppure un meno (-) unario.

## Operatori alfabetici

È possibile concatenare i caratteri.

### Esempi

Negli esempi che seguono, cognome è 'BIANCHI' e nome è 'FABIO':

Operatore	Concatenazione	Esempio	Risultato
+	Mantiene i caratteri finalivuoti	nome + cognome	'FABIO BIANCHI'
-	Sposta i caratteri vuoti alla fine	nome - cognome	'FABIOBIANCHI'

## Operatori data

È possibile modificare le date.

### Esempi

Nei seguenti esempi, data\_assunzione è {D "2008-30-01"}.

Operatore	Effetto sulla data	Esempio	Risultato
+	Aggiunge un numero di giorni ad una data	data_assunzione + 5	{D '2008-02-04'}
-	Trova il numero di giorni tra due date o sottrae un numero di giorni da una data	data_assunzione - {D '2008-01-01'} data_assunzione - 10	29 {D '2008-01-20'}

Altri esempi:

```
SELECT Data_Vendita, Data_Vendita + 30 AS agg FROM Dati_Vendite
SELECT Data_Vendita, Data_Vendita - 30 AS agg FROM Dati_Vendite
```

## Operatori relazionali

Operatore	Significato
=	Uguale
<>	Diverso da
>	Maggiore di
>=	Maggiore o uguale a
<	Minore di
<=	Minore o uguale a
LIKE	Corrisponde ad una struttura
NOT LIKE	Non corrisponde ad una struttura
IS NULL	Uguale a NULL
IS NOT NULL	Diverso da NULL
BETWEEN	Intervallo di valori tra un limite inferiore e un limite superiore
IN	Un membro di un gruppo di valori specificati o un membro di una subquery
NOT IN	Non un membro di un gruppo di valori specificati né un membro di una subquery
EXISTS	'Vero' se una subquery ha restituito almeno un record
ANY	Confronta un valore con ogni valore restituito da una subquery (l'operatore deve essere preceduto da =, <>, >, >=, <, o <=); =Any equivale a In
ALL	Confronta un valore con ogni valore restituito da una subquery (l'operatore deve essere preceduto da =, <>, >, >=, <, o <=)

### Esempi

```
SELECT Dati_Vendite.ID_Fattura FROM Dati_Vendite
WHERE Dati_Vendite.ID_Venditore = ' SP-1'
SELECT Dati_Vendite.Quantità FROM Dati_Vendite WHERE Dati_Vendite.ID_Fattura
<> 125
```

```

SELECT Dati_Vendite.Quantità FROM Dati_Vendite WHERE Dati_Vendite.Quantità
> 3000
SELECT Dati_Vendite.Ora_Vendita FROM Dati_Vendite
WHERE Dati_Vendite.Ora_Vendita < '12:00:00'
SELECT Dati_Vendite.Nome_Società FROM Dati_Vendite
WHERE Dati_Vendite.Nome_Società LIKE '%University'
SELECT Dati_Vendite.Nome_Società FROM Dati_Vendite
WHERE Dati_Vendite.Nome_Società NOT LIKE '%University'
SELECT Dati_Vendite.Quantità FROM Dati_Vendite WHERE Dati_Vendite.Quantità
IS NULL
SELECT Dati_Vendite.Quantità FROM Dati_Vendite WHERE Dati_Vendite.Quantità
IS NOT NULL
SELECT Dati_Vendite.ID_Fattura FROM Dati_Vendite
WHERE Dati_Vendite.ID_Fattura BETWEEN 1 AND 10
SELECT COUNT(Dati_Vendite.ID_Fattura) AS agg
FROM Dati_Vendite WHERE Dati_Vendite.INVOICE_ID IN (50,250,100)
SELECT COUNT(Dati_Vendite.ID_Fattura) AS agg
FROM Dati_Vendite WHERE Dati_Vendite.INVOICE_ID NOT IN (50,250,100)
SELECT COUNT(Dati_Vendite.ID_Fattura) AS agg FROM Dati_Vendite
WHERE Dati_Vendite.INVOICE_ID NOT IN (SELECT Dati_Vendite.ID_Fattura
FROM Dati_Vendite WHERE Dati_Vendite.ID_Venditore = 'SP-4')
SELECT*
FROM Dati_Vendite WHERE EXISTS (SELECT Dati_Vendite.Quantità
FROM Dati_Vendite WHERE Dati_Vendite.ID_Venditore IS NOT NULL)
SELECT*
FROM Dati_Vendite WHERE Dati_Vendite.Quantità = ANY (SELECT
Dati_Vendite.Quantità
FROM Dati_Vendite WHERE Dati_Vendite.ID_Venditore = 'SP-1')
SELECT*
FROM Dati_Vendite WHERE Dati_Vendite.Quantità = ALL (SELECT
Dati_Vendite.Quantità
FROM Dati_Vendite WHERE Dati_Vendite.ID_Venditore IS NULL)

```

## Operatori logici

È possibile combinare due o più condizioni. Le condizioni devono essere correlate con AND o OR, come ad esempio:

```
stipendio = 40000 AND detrazioni = 1
```

L'operatore logico NOT è utilizzato per invertire il significato, come ad esempio:

```
NOT (stipendio = 40000 AND detrazioni = 1)
```

## Esempi

```

SELECT* FROM Dati_Vendite WHERE Dati_Vendite.Nome_Società
NOT LIKE '%University' AND Dati_Vendite.Quantità > 3000
SELECT* FROM Dati_Vendite WHERE (Dati_Vendite.Nome_Società
LIKE '%University' OR Dati_Vendite.Quantità > 3000)
AND Dati_Vendite.ID_Venditore = 'SP-1'

```

## Funzioni

FileMaker SQL supporta molte funzioni utilizzabili nelle espressioni. Alcune funzioni restituiscono stringhe di caratteri, alcune numeri, altre date.

### Funzioni che restituiscono stringhe di caratteri

Funzioni che restituiscono stringhe di caratteri	Descrizione	Esempio
CHR	Converte un codice ASCII in una stringa da un carattere	CHR(67) restituisce C
CURRENT_USER	Restituisce l'ID di accesso specificato al momento della connessione	
DAYNAME	Restituisce il nome del giorno che corrisponde a una data specificata.	
RTRIM	Rimuove gli spazi finali da una stringa	RTRIM(' ABC ') restituisce ' ABC'
TRIM	Rimuove gli spazi vuoti iniziali e finali da una stringa	TRIM(' ABC ') restituisce 'ABC'
LTRIM	Rimuove gli spazi iniziali da una stringa	LTRIM(' ABC') restituisce 'ABC'
UPPER	Trasforma tutte le lettere di una stringa in maiuscole	UPPER('Allen') restituisce 'ALLEN'
LOWER	Trasforma tutte le lettere di una stringa in minuscole	LOWER('Allen') restituisce 'allen'
LEFT	Restituisce i caratteri più a sinistra di una stringa	LEFT('Mattson',3) restituisce 'Mat'
MONTHNAME	Restituisce i nomi dei mesi di calendario.	
RIGHT	Restituisce i caratteri più a destra di una stringa	RIGHT('Mattson',4) restituisce 'tson'
SUBSTR SUBSTRING	Restituisce una sottostringa di una stringa, con i parametri della stringa, il primo carattere da estrarre e il numero di caratteri da estrarre (opzionale)	SUBSTR('Conrad',2,3) restituisce 'onr' SUBSTR('Conrad',2) restituisce 'onrad'
SPACE	Genera una stringa di spazi vuoti	SPACE(5) restituisce ' '
STRVAL	Converte un valore di qualsiasi tipo in una stringa di caratteri	STRVAL('Woltman') restituisce 'Woltman' STRVAL(5 * 3) restituisce '15' STRVAL(4 = 5) restituisce 'Falso' STRVAL({D ' 2008-12-25'}) restituisce "2008-12-25"
TIME TIMEVAL	Restituisce l'ora del giorno sotto forma di stringa	Alle 21:49, TIME() restituisce 21:49:00
USERNAME USER	Restituisce l'ID di accesso specificato al momento della connessione	

**Nota** La funzione TIME() non è più in uso. Al suo posto, utilizzare la funzione standard SQL CURRENT\_TIME.

### Esempi

```
SELECT CHR(67) + SPACE(1) + CHR(70) FROM Venditori
SELECT RTRIM(' ' + Venditori.ID_Venditore) AS agg FROM Venditori
SELECT TRIM(SPACE(1) + Venditori.ID_Venditore) AS agg FROM Venditori
SELECT LTRIM(' ' + Venditori.ID_Venditore) AS agg FROM Venditori
SELECT UPPER(Venditori.Venditore) AS agg FROM Venditori
```

```

SELECT LOWER(Venditori.Venditore) AS agg FROM Venditori
SELECT LEFT(Venditori.Venditore, 5) AS agg FROM Venditori
SELECT RIGHT(Venditori.Venditore, 7) AS agg FROM Venditori
SELECT SUBSTR(Venditori.ID_Venditore, 2, 2) + SUBSTR(Venditori.ID_Venditore,
4, 2) AS agg FROM Venditori
SELECT SUBSTR(Venditori.Salesperson_ID, 2) + SUBSTR(Venditori.ID_Venditore,
4) AS agg FROM Venditori
SELECT SPACE(2) + Venditori.ID_Venditore AS ID_Venditore FROM Venditori
SELECT STRVAL('60506') AS agg FROM Dati_Vendite WHERE Dati_Vendite.Fattura
= 1

```

## Funzioni che restituiscono numeri

### Funzioni che restituiscono numeri

Funzioni che restituiscono numeri	Descrizione	Esempio
ABS	Restituisce il valore assoluto dell'espressione numerica	
ATAN	Restituisce l'arcotangente dell'argomento come angolo espresso in radianti	
ATAN2	Restituisce l'arcotangente delle coordinate x e y come angolo espresso in radianti	
B	Restituisce l'equivalente decimale di un numero binario	B '1001' restituisce 9
CEIL CEILING	Restituisce il valore intero più piccolo, maggiore o uguale all'argomento	
DEG DEGREES	Restituisce il numero di gradi dell'argomento che è un angolo espresso in radianti	
DAY	Restituisce il giorno di una data	DAY({d '2010/01/30'}) restituisce 30
DAYOFWEEK	Restituisce il giorno della settimana (1-7) di un'espressione di data	DAYOFWEEK({d '2004/05/01'}) restituisce 7
MOD	Divide due numeri e restituisce il resto ottenuto dalla divisione	MOD(10,3) restituisce 1
EXP	Restituisce un valore che è la base del logaritmo naturale (e) elevato alla potenza specificata dall'argomento	
FLOOR	Restituisce il valore intero più elevato, inferiore o uguale all'argomento	
hour	Restituisce la parte dell'ora di un valore.	
INT	Restituisce la parte intera di un numero	INT(6.4321) restituisce 6
LEN LENGTH	Restituisce la lunghezza di una stringa	LEN('ABC') restituisce 3
MONTH	Restituisce il mese di una data	MONTH({d '2010/01/30'}) restituisce 1
LN LOG	Restituisce il logaritmo naturale dell'argomento	
MAX	Restituisce il maggiore di due numeri	MAX(66,89) restituisce 89
MIN	Restituisce il minore di due numeri	MIN(66,89) restituisce 66
MINUTE	Restituisce la parte dei minuti di un valore	

**Funzioni che restituiscono numeri**

	<b>Descrizione</b>	<b>Esempio</b>
NUMVAL	Converte una stringa di caratteri in un numero; se la stringa di caratteri non è un numero valido, restituisce 0	NUMVAL ('123') restituisce 123
PI	Restituisce il valore della costante matematica pi	
POW	Eleva un numero ad una potenza	POW (7, 2) restituisce 49
RADIANS	Restituisce il numero di radianti di un argomento espresso in gradi	
ROUND	Arrotonda un numero	ROUND (123.456, 0) restituisce 123 ROUND (123.456, 2) restituisce 123.46 ROUND (123.456, -2) restituisce 100
SECOND	Restituisce la parte dei secondi di un valore	
SIGN	Un indicatore del segno dell'argomento: -1 per negativo, 0 per 0 e 1 per positivo.	
SIN	Restituisce il seno dell'argomento	
SQRT	Restituisce la radice quadrata dell'argomento	
TAN	Restituisce la tangente dell'argomento	
VAL	Converte una stringa di caratteri in un numero; se la stringa di caratteri non è un numero valido, restituisce 0	VAL ('123') restituisce 123
X	Restituisce l'equivalente decimale di un numero esadecimale	X 'b9' restituisce 185
YEAR	Restituisce l'anno di una data	YEAR ({d '2010/01/30'}) restituisce 2010

**Funzioni che restituiscono date****Funzioni che restituiscono date**

	<b>Descrizione</b>	<b>Esempio</b>
CURDATE CURRENT_DATE	Restituisce la data di oggi	
CURTIME CURRENT_TIME	Restituisce l'ora corrente	
CURTIMESTAMP CURRENT_TIMESTAMP TIMESTAMPVAL	Restituisce il valore corrente dell'indicatore data e ora	
DATE TODAY	Restituisce la data di oggi	Se la data odierna è 11/21/2010, DATE () restituisce 2010-11-21
DATEVAL	Converte una stringa di caratteri in una data	DATEVAL ("01/30/2011") restituisce 2011-01-30

**Nota** La funzione DATE() non è più in uso. Al suo posto, utilizzare la funzione standard SQL CURRENT\_DATE.

## Precedenza degli operatori

Più le espressioni sono complesse, più l'ordine con cui le espressioni vengono valutate è importante. Questa tabella mostra l'ordine in cui sono valutati gli operatori. Gli operatori nella prima linea sono valutati per primi, e così via. Gli operatori sulla stessa riga vengono valutati da sinistra a destra nell'espressione.

Precedenza	Operatore
1	'-' unario, '+' unario
2	^, **
3	*, /
4	+, -
5	=, <>, <, <=, >, >=, Like, Not Like, Is Null, Is Not Null, Between, In, Exists, Any, All
6	Not
7	AND
8	OR

Nell'esempio che segue viene mostrata l'importanza della precedenza:

```
WHERE stipendio > 40000 OR data_assunzione > {d '2008/01/30'} AND sett = 'D101'
```

Poiché AND viene valutato per primo, questa query recupera i dipendenti nel reparto D101 assunti dopo il 30.01.08, e tutti i dipendenti con uno stipendio superiore a 40.000 Euro, indipendentemente dal settore e dalla data di assunzione.

Per forzare la clausola in modo che venga valutata in un ordine diverso, usare le parentesi e racchiudere le condizioni da valutare per prime. Ad esempio:

```
WHERE (stipendio > 40000 OR data_assunzione > {d '2008/01/30'}) AND sett = 'D101'
```

Recupera i dipendenti nel settore D101 con uno stipendio superiore a 40.000 Euro o assunti dopo il 30.01.08.

## Funzioni di Catalogo ODBC

Il driver client ODBC supporta le seguenti funzioni di Catalogo:

- SQLTables - le informazioni di catalogo sono memorizzate e riportate come nomi di parti singole (solo nome tabella).
- SQLColumns
- SQLColumnPrivileges
- SQLDescribeCol
- SQLGetTypeInfo

## Funzioni Meta Data JDBC

Il driver client JDBC supporta le seguenti funzioni Meta Data:

- `getColumns`
- `getColumnPrivileges`
- `getMetaData`
- `getTypeInfo`
- `getTables`
- `getTableTypes`

## Parole chiave SQL riservate

La seguente tabella elenca le parole chiave riservate che non devono essere utilizzate come nomi di colonne, tabelle, alias o altri oggetti definiti dall'utente. Se vengono segnalati errori di sintassi, è possibile che sia stata utilizzata una di queste parole riservate. Se si vuole utilizzare una di queste parole chiave, è necessario utilizzare le virgolette per evitare che queste siano considerate come parole chiave.

Ad esempio, la seguente istruzione Create Table illustra come utilizzare la parola chiave "OID" come un nome elemento dati.

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

ABSOLUTE	COMMIT	DOUBLE	INSERT	OF
ACTION	CONNECT	DROP	INT	ON
ADD	CONNECTION	ELSE	INTEGER	ONLY
ALL	CONSTRAINT	END	INTERSECT	OPEN
ALLOCATE	CONSTRAINTS	END_EXEC	INTERVAL	OPTION
ALTER	CONTINUE	ESCAPE	INTO	OR
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	ROWS
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	USAGE
SET	USER
SIZE	USERNAME
SMALLINT	USING
SOME	VALUE
SPACE	VALUES
SQL	VARBINARY
SQLCODE	VARCHAR
SQLERROR	VARYING
SQLSTATE	VIEW
STRVAL	WHEN
SUBSTRING	WHENEVER
SUM	WHERE
SYSTEM_USER	WITH
TABLE	WORK
TEMPORARY	WRITE
THEN	YEAR
TIME	ZONE
TIMESTAMP	
TIMESTAMPVAL	
TIMEVAL	
TIMEZONE_HOUR	
TIMEZONE_MINUTE	
TO	
TODAY	
TRAILING	
TRANSACTION	
TRANSLATE	
TRANSLATION	
TRIM	
TRUE	
UNION	
UNIQUE	
UNKNOWN	
UPDATE	
UPPER	
USAGE	
USER	
USERNAME	
USING	



# Capitolo 8

## Informazioni di riferimento

### Mappatura di campi FileMaker su dati di tipo ODBC

Nella tabella che segue è illustrato il modo in cui i campi FileMaker vengono mappati sui dati di tipo ODBC standard.

tipo di campo FileMaker	Converte nel tipo di dati ODBC	Informazioni sul tipo di dati
testo	SQL_VARCHAR	La lunghezza massima di una colonna è di 1 milione di caratteri, a meno che non si specifichi un Numero massimo di caratteri inferiore per il campo di testo di FileMaker. FileMaker restituisce le stringhe vuote come NULL.
numero	SQL_DOUBLE	Il tipo campo Numero di FileMaker può contenere valori positivi o negativi, da $10^{-308}$ fino a $10^{+308}$ , con max. 15 cifre significative.
data	SQL_DATE	
ora	SQL_TIME	Il tipo di campo Ora FileMaker può contenere l'ora del giorno oppure un intervallo di tempo. L'intervallo di tempo viene restituito come un'ora del giorno, a meno non che sia inferiore a 0 o superiore a 24 ore (in entrambi i casi viene restituito un valore di 0).
indicatore data e ora	SQL_TIMESTAMP	
contenitore (BLOB)	SQL_LONGVARBINARY	Da un campo Contenitore si possono recuperare dati binari, informazioni di riferimento al file o dati di un tipo specifico di file. All'interno di un'istruzione SELECT utilizzare la funzione CAST per recuperare le informazioni di riferimento al file e la funzione RicavaCome per recuperare i dati di un tipo specifico di file.
calcolo		Il risultato viene mappato sul tipo di dati ODBC corrispondente.

La lunghezza delle stringhe è opzionale nelle dichiarazioni di tabella. Tutte le stringhe vengono memorizzate e recuperate in Unicode.

**Nota** In FileMaker i campi multipli sono supportati come le matrici. Esempi:

```
INSERT INTO mytable(repField[3]) VALUES ('this is rep 3')
SELECT repField[1], repField[2] FROM mytable
```

## Mappatura di campi FileMaker su dati di tipo JDBC

Il driver client JDBC utilizza le seguenti mappature per convertire i dati di tipo FileMaker in dati JDBC SQL. (Per informazioni su questi tipi, vedere le pagine Web di documentazione di JDK 1,5 all'indirizzo [www.javasoft.com](http://www.javasoft.com)).

### tipo di campo FileMaker    Converte in tipo JDBC SQL

text	java.sql.Types.VARCHAR
numero	java.sql.Types.DOUBLE
date	java.sql.Types.DATE
ora	java.sql.Types.TIME
Indicatore data e ora	java.sql.Types.TIMESTAMP
Contenitore	java.sql.Types.BLOB
Calcolo	specificato dal tipo di dati del risultato del calcolo

Il driver client JDBC converte i dati FileMaker di tipo Calcolo nel tipo JDBC SQL che corrisponde al risultato del calcolo. Ad esempio, il driver client JDBC converte un calcolo FileMaker che produce dati di tipo Indicatore data e ora (timestamp) in `java.sql.Types.TIMESTAMP`.

## Messaggi di errore ODBC e JDBC

Di seguito sono presentati i formati di base dei messaggi di errore che possono essere visualizzati durante l'uso di FileMaker e ODBC/JDBC.

### Messaggi di errore ODBC

I messaggi di errore possono essere generati da:

- Errori driver ODBC
- Errori di FileMaker e FileMaker xDBC Listener

### Messaggi di errore ODBC FileMaker

Un errore verificatosi in FileMaker Listener o nell'origine dati specifica il nome dell'origine dati, utilizzando il seguente formato:

[FileMaker] [ODBC FileMaker] messaggio

Ad esempio l'origine dati FileMaker potrebbe generare il seguente messaggio:

[FileMaker] [ODBC FileMaker] Nome utente/password non validi

Se si riceve un errore di questo tipo, significa che è stata eseguita un'operazione non valida relativamente al sistema di database. Per ulteriori informazioni consultare la documentazione FileMaker oppure contattare l'amministratore del database.

Messaggi consecutivi di errore in colonne diverse possono visualizzare a volte un nome di colonna non corretto.

## Messaggi di errore JDBC

Il driver FileMaker JDBC genera errori per l'applicazione chiamante restituendo SQLExceptions. I messaggi di errore possono essere generati da:

- Errori driver JDBC
- Errori di FileMaker e FileMaker xDBC Listener

### Messaggi di errore JDBC FileMaker

Un errore verificatosi in FileMaker Listener o nell'origine dati specifica il nome dell'origine dati, utilizzando il seguente formato:

[FileMaker] [FileMaker JDBC] messaggio

Ad esempio l'origine dati FileMaker potrebbe generare il seguente messaggio:

[FileMaker] [JDBC FileMaker] Nome utente/password non validi

Se si riceve un errore di questo tipo, significa che è stata eseguita un'operazione non valida relativamente al sistema di database. Per ulteriori informazioni consultare la documentazione FileMaker oppure contattare l'amministratore del database.



# Indice

## A

- accesso mediante il privilegio esteso ODBC/JDBC 22
- accesso remoto 9
- account e privilegi 22
- aggiornamenti ed eliminazioni nella posizione 41
- alias colonna 38
- alias tabella 38, 39
- ALTER TABLE (Istruzione SQL) 47
- Amministratore ODBC (Mac OS) 26
- Amministrazione origine dati ODBC (Windows) 25
- applicazione client, utilizzo di FileMaker come 7

## C

- campi
  - mappatura JDBC 62
  - mappatura ODBC 61
- campi multipli 61
- campo Contenitore
  - con istruzione INSERT 44
  - con istruzione SELECT 42
  - con istruzione UPDATE 46
  - Mappatura del tipo di dati JDBC 62
  - Mappatura tipo di dati ODBC 61
- caratteri vuoti 50
- chiavi generate automaticamente 31
- collegamento 39
- condivisione, impostazione ODBC/JDBC 22
- configurazione di un'origine dati FileMaker
  - attraverso ODBC (Mac OS) 25
  - attraverso ODBC (Windows) 23
  - via JDBC 34
- conformità standard 37
- conformità standard SQL 37
- connessioni database, numero supportato 9
- connessioni, database 9
- costanti in espressioni SQL 49
- CREATE INDEX (Istruzione SQL) 47
- CREATE TABLE (Istruzione SQL) 46
- cursore aperto 32
- cursori
  - in JDBC 32
  - in ODBC 41

## D

- database, DSN 24
- dati binari
  - uso in SELECT 42
- DELETE (Istruzione SQL) 44
- disattivazione di un file di database FileMaker condiviso 9
- driver
  - disinstallazione vecchi 10
- Driver client JDBC
  - classe e punto di ingresso principale del driver 32
  - funzioni meta data 57
  - mappatura tipi dati 62
  - portali 37
  - registrazione nella gestione driver JDBC 32
  - specificazione dell'URL JDBC 32
  - Supporto Unicode 37
  - verifica accesso 35
- Driver client ODBC
  - funzioni Catalogo 56
  - mappatura tipi dati 61
  - portali 37
  - Supporto Unicode 37
  - verifica accesso (Mac OS) 26
  - verifica accesso (Windows) 24
- DROP INDEX (Istruzione SQL) 48
- DSN
  - creazione (Mac OS) 25
  - creazione (Windows) 23
  - uno per file 10

## E

- errori di sintassi 57
- espressioni in SQL 49
- espressioni SQL 49
  - costanti 49
  - funzioni 53
  - letterali 49
  - nomi campi 49
  - notazione esponenziale o scientifica 50
  - operatori caratteri 50
  - operatori di data 51
  - operatori logici 52
  - operatori numerici 50
  - operatori relazionali 51
  - precedenza operatori 56

**F**

## file

- impostazione dell'accesso a 22
- organizzazione su un computer 9
- uso in campi Contenitore 43
- file bitmap in campi Contenitore 43
- file di immagine in campi Contenitore 43
- File Quicktime in campi Contenitore 43
- FOR UPDATE (clausola SQL) 41
- formati data 49
- formati dei messaggi di errore 62
- formati dell'indicatore data e ora 49
- formati ora 49
- FROM (clausola SQL) 39
- FULL OUTER JOIN 39
- funzione ABS 54
- funzione ATAN 54
- funzione ATAN2 54
- Funzione B 54
- funzione CAST 43, 61
- funzione CEIL 54
- funzione CEILING 54
- funzione CHR 53
- funzione CURDATE 55
- funzione CURRENT\_USER 53
- funzione CURRENT\_DATE 55
- funzione CURRENT\_TIME 55
- funzione CURRENT\_TIMESTAMP 55
- funzione CURRENT\_USER 53
- funzione CURTIME 55
- funzione CURTIMESTAMP 55
- funzione DATE 55
- funzione DATEVAL 55
- funzione DAY 54
- funzione DAYNAME 53
- funzione DAYOFWEEK 54
- funzione DEG 54
- funzione DEGREES 54
- funzione EXP 54
- funzione FLOOR 54
- funzione HOUR 54
- funzione INF 53
- funzione INT 54
- funzione LEFT 53
- funzione LEN 54
- funzione LN 54
- funzione LOG 54

- funzione LTRIM 53
- funzione MAX 54
- funzione MIN 54
- funzione MINUTE 54
- funzione MOD 54
- funzione MONTH 54
- funzione MONTHNAME 53
- funzione NUMVAL 55
- funzione PI 55
- funzione POW 55
- funzione RADIANS 55
- funzione RicavaCome 43, 61
- funzione RIGHT 53
- funzione ROUND 55
- funzione RTRIM 53
- funzione SECOND 55
- funzione SIGN 55
- funzione SIN 55
- funzione SPACE 53
- funzione SQRT 55
- funzione STRVAL 53
- funzione SUBSTR 53
- funzione SUBSTRING 53
- funzione TAN 55
- funzione TIME 53
- funzione TIMESTAMPVAL 55
- funzione TIMEVAL 53
- funzione TODAY 55
- funzione TRIM 53
- funzione UPPER 53
- funzione USERNAME 53
- funzione VAL 55
- funzione X 55
- funzione YEAR 55
- funzioni aggregate in SQL 48
- Funzioni aggregate SQL 48
- funzioni Catalogo per ODBC 56
- funzioni meta data per JDBC 57
- funzioni nelle espressioni SQL 53
- funzioni stringa 53

**G**

- GROUP BY (clausola SQL) 40

**H**

- HAVING (clausola SQL) 40

host, DSN 23, 25

## I

INNER JOIN 39

INSERT (Istruzione SQL) 44

introduzione

impostazione di privilegi e condivisione 22

utilizzo di ODBC e JDBC con FileMaker 7

Istruzioni SQL

ALTER TABLE 47

CREATE INDEX 47

CREATE TABLE 46

DELETE 44

DROP INDEX 48

INSERT 44

parole chiave riservate 57

SELECT 37

supportate da driver client 37

UPDATE 45

## J

JDBC

descrizione 31

driver client, descrizione 31

messaggi di errore 63

uso 7

JDBC SPI 32

JDK (Java Development Kit) 32

## L

LEFT JOIN 39

LEFT OUTER JOIN 39

letterali nelle espressioni SQL 49

## M

Mac OS

Creazione di un DSN 25

Requisiti per driver client JDBC 29

Requisiti per driver client ODBC 15

verifica dell'accesso ODBC 26

mappatura tipi dati

Driver client JDBC 62

Driver client ODBC 61

## N

nomi colonna 23

nomi dei campi nelle espressioni SQL 49

nomi origine dati. *Vedere* DSN

NOT NULL (clausola SQL) 46

notazione esponenziale in espressioni SQL 50

notazione scientifica nelle espressioni SQL 50

## O

ODBC

campi multipli 61

conformità standard 37

descrizione 21

messaggi di errore 62

uso 7

Operatore ALL 51

operatore AND 52

Operatore ANY 51

Operatore BETWEEN 51

operatore DISTINCT 38

operatore EXISTS 51

operatore IN 51

operatore IS NOT NULL 51

operatore IS NULL 51

Operatore LIKE 51

operatore NOT 52

operatore NOT IN 51

operatore NOT LIKE 51

operatore OR 52

operatori caratteri nelle espressioni SQL 50

operatori di data nelle espressioni SQL 51

operatori logici nelle espressioni SQL 52

operatori numerici nelle espressioni SQL 50

operatori relazionali nelle espressioni SQL 51

ORDER BY (clausola SQL) 41

origine dati

configurazione accesso tramite JDBC 34

configurazione dell'accesso attraverso ODBC

(Mac OS) 25

configurazione dell'accesso attraverso ODBC

(Windows) 23

disattivazione di un file di database FileMaker  
condiviso 9

un DSN per ogni file di database FileMaker 10

verifica accesso attraverso ODBC (Mac OS) 26

verifica accesso attraverso ODBC (Windows) 24

verifica accesso tramite JDBC 35

Origine dati server 26

OUTER JOIN 39

## P

parole chiave SQL riservate 57

- parole chiave, SQL riservate 57
- password
  - con JDBC 34
  - con ODBC 24, 26
- porta, impostazione per JDBC 30
- portali 37
- precedenza operatori nelle espressioni SQL 56
- PREVENT INDEX CREATION 48
- privilegi estesi 22
- privilegi, estesi 22
- prodotti FileMaker 9
- proprietà del driver
  - Driver client JDBC 34
  - driver client ODBC (Mac OS) 25
  - Driver client ODBC (Windows) 23

## R

- Registrazione del driver client JDBC 32
- requisiti di sistema 15, 29
- requisiti per l'installazione 15, 29
- rete, requisiti 10
- RIGHT JOIN 39
- RIGHT OUTER JOIN 39

## S

- SELECT (Istruzione SQL) 37
  - dati binari 42
  - stringa vuota 42
  - tipo di dati BLOB 42
- SQL-92 37
- SQLExceptions 63
- stringa vuota
  - uso in SELECT 42
- strumenti RAD (Rapid Application Development) 31
- subquery 44
- Supporto del punto di salvataggio 31
- Supporto Unicode 37

## T

- testing accesso
  - Driver client JDBC 35
  - driver client ODBC (Mac OS) 26
  - Driver client ODBC (Windows) 24
- tipi di dati, mappatura
  - Driver client JDBC 62
  - Driver client ODBC 61
- tipo di dati 32

- tipo di dati ARRAY 32
- tipo di dati BLOB
  - uso in CREATE TABLE 46
  - uso in SELECT 42
- tipo di dati booleano 32
- tipo di dati CLOB 32
- tipo di dati DATALINK 32
- tipo di dati SQL\_C\_WCHAR 37

## U

- UNION (operatore SQL) 40
- UPDATE (Istruzione SQL) 45
- URL (Uniform Resource Locator) per il driver client JDBC 32

## V

- valore Null 44, 61
- valore vuoto nelle colonne 44
- VALUES (clausola SQL) 44
- verifica accesso
  - Driver client JDBC 35
  - driver client ODBC (Mac OS) 26
  - Driver client ODBC (Windows) 24
- versione Java 29

## W

- WHERE (clausola SQL) 39
- Windows
  - Creazione di un DSN 23
  - Requisiti per driver client JDBC 29
  - Requisiti per driver client ODBC 15
  - verifica dell'accesso ODBC 24