

Controlador ODBC para o dBASE

Para todos os usuários

Os tópicos seguintes descrevem o controlador ODBC para o dBASE e sua instalação.

[Visão geral](#)

[Requerimentos de hardware e software](#)

[Configurando o controlador ODBC para o dBASE](#)

[Adicionando, modificando e excluindo uma origem de dados dBASE](#)

[Conectando-se a uma origem de dados dBASE](#)

[Usando o controlador ODBC para o dBASE](#)

Para usuários experientes

Os tópicos seguintes descrevem como usar diretamente o controlador ODBC para o dBASE.

[Seqüências de conexão \(Avançado\)](#)

[Instruções SQL \(Avançado\)](#)

[Tipos de dados \(Avançado\)](#)

[Mensagens de erro \(Avançado\)](#)

Para programadores

Os tópicos seguintes fornecem informações de programação sobre o Controlador ODBC para o dBASE. Elas são dirigidas a programadores de aplicativos e requerem o conhecimento da interface Open Database Connectivity ([ODBC](#)) e da Interface de Programas Aplicativos ([API](#)).

[Valores retornados SQLGetInfo \(Programação\)](#)

[Funções API do ODBC \(Programação\)](#)

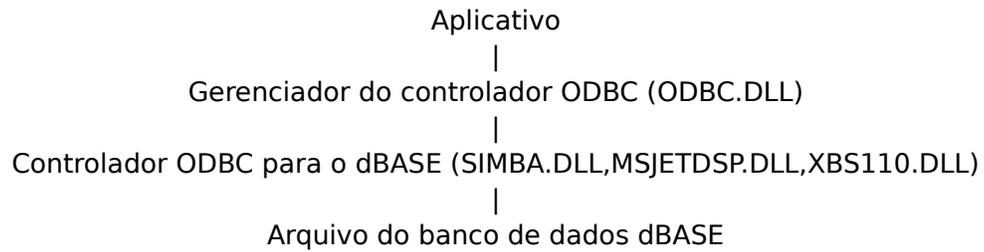
[Tópicos de implementação \(Programação\)](#)

Visão geral

Ver também

O controlador ODBC para o dBASE permite a você abrir e consultar o banco de dados dBASE através da interface Open Database Connectivity ([ODBC](#))

A arquitetura do aplicativo/controlador é:



Ver também

Para todos os usuários

[Adicionando, modificando e excluindo uma origem de dados dBASE](#)

[Conectando-se a uma origem de dados dBASE](#)

[Requerimentos de hardware e software](#)

[Configurando o controlador ODBC para o dBASE](#)

[Usando o controlador ODBC para o dBASE](#)

Requerimentos de hardware e software

Ver também

Para acessar os dados do dBASE você precisa ter:

- Controlador ODBC para o dBASE.
- Arquivos de banco de dados dBASE III ou dBASE IV.
- Computador executando MS-DOS 3.3 ou posterior.
- Microsoft Windows 3.0a ou posterior.
- Gerenciador do controlador ODBC 1.0 (ODBC.DLL).

Para adicionar, modificar ou excluir controladores ou origens de dados, você deve ter a opção ODBC no Painel de Controle (ou o programa Administrador ODBC, caso esteja executando o Microsoft Windows 3.0a) instalada em seu computador.

Para maiores informações sobre os bancos de dados dBASE, consulte a documentação do dBASE.

Ver também

Para todos os usuários

[Configurando o controlador ODBC para o dBASE](#)

Configurando o controlador ODBC para o dBASE

Ver também

Para configurar o controlador ODBC para o dBASE

- 1 No grupo Principal da janela Gerenciador de Programas, clique duas vezes no ícone Painel de Controle. Na janela Painel de Controle, clique duas vezes no ícone ODBC.

Observação No Microsoft Windows 3.0a, inicie o Administrador ODBC clicando duas vezes no ícone Administrador ODBC no grupo Microsoft ODBC.

- 2 Na caixa de diálogo **Origens de Dados**, escolha o botão "Controladores".
- 3 Na caixa de diálogo **Controladores**, escolha o botão "Adicionar".
- 4 Na caixa de diálogo **Adicionar Controlador**, insira na caixa de texto o nome da unidade de disco e do diretório contendo o controlador ODBC para o dBASE. Ou escolha o botão "Procurar" para selecionar uma unidade de disco e o nome de diretório.
- 5 Escolha o botão "OK".
- 6 Na caixa de diálogo **Instalar Controladores**, selecione dBASE na lista "Controladores ODBC disponíveis".
- 7 Escolha o botão "OK" para instalar o controlador.

Observação O controlador ODBC para o dBASE pode compartilhar algumas bibliotecas de vínculo dinâmico (DLL) com outros controladores instalados em seu computador. Neste caso, você será orientado a sobrescrever o controlador ODBC para o dBASE, tendo este sido instalado ou não. Escolha o botão "Sim" para instalar o controlador.

Antes de usar o controlador, você deve adicionar uma origem de dados para ele. Além disso, deve incluir a seguinte linha em seu arquivo AUTOEXEC.BAT:

```
SHARE /L:200
```

O comando SHARE permite compartilhar e proteger arquivos em ambiente multitarefa. A opção /L especifica o número de arquivos que podem ser bloqueados de uma só vez.

Para excluir o controlador ODBC para o dBASE

- 1 No grupo Principal da janela Gerenciador de Programas, clique duas vezes no ícone Painel de Controle. Na janela Painel de Controle, clique duas vezes no ícone ODBC.

Observação No Microsoft Windows 3.0a, inicie o Administrador ODBC clicando duas vezes no ícone Administrador ODBC no grupo Microsoft ODBC.

- 2 Na caixa de diálogo **Origens de Dados**, escolha o botão "Controladores".
- 3 Na caixa de diálogo **Controladores**, selecione o controlador ODBC para o dBASE na lista "Controladores do ODBC instalados".
- 4 Escolha o botão "Excluir".
O programa de configuração ODBC para o dBASE pergunta se você quer remover o controlador e todas as origens de dados que utilizam o controlador.
- 5 Escolha o botão "Sim".

Ver também

Para todos os usuários

[Adicionando, modificando e excluindo uma origem de dados dBASE](#)
[Requerimentos de hardware e software](#)

Adicionando, modificando e excluindo uma origem de dados dBASE

Ver também

Antes de acessar dados com o controlador ODBC para o dBASE, você deve adicionar uma origem de dados para ele. O controlador ODBC para o dBASE usa esta informação para acessar os dados. O usuário pode mudar ou excluir uma origem de dados a qualquer hora.

Para adicionar uma origem de dados dBASE

- 1 No grupo Principal da janela Gerenciador de Programas clique duas vezes no ícone Painel de Controle. Na janela Painel de Controle, clique duas vezes no ícone ODBC.

Observação No Microsoft Windows 3.0a, inicie o Administrador ODBC clicando duas vezes no ícone Administrador ODBC, no grupo Microsoft ODBC.

- 2 Na caixa de diálogo **Origens de Dados**, escolha o botão "Adicionar".
- 3 Na caixa de diálogo **Adicionar Origem de Dados**, selecione "dBASE" na lista "Controladores ODBC instalados" e escolha "OK".
- 4 Na caixa de diálogo Configuração do ODBC para o dBASE, insira a informação para configurar a origem de dados.

Para modificar uma origem de dados dBASE

- 1 No grupo Principal da janela Gerenciador de Programas, clique duas vezes no ícone Painel de Controle. Na janela Painel de Controle, clique duas vezes no ícone ODBC.

Observação No Microsoft Windows 3.0a, inicie o Administrador ODBC clicando duas vezes no ícone Administrador ODBC, no grupo Microsoft ODBC.

- 2 Na caixa de diálogo **Origens de Dados**, selecione a origem de dados na lista "Origens de dados".
- 3 Escolha o botão "Config".
- 4 Na caixa de diálogo Configuração do ODBC para o dBASE, insira a informação para configurar a origem de dados.

Para excluir uma origem de dados dBASE

- 1 No grupo Principal da janela Gerenciador de Programas, clique duas vezes no ícone Painel de Controle. Na janela Painel de Controle, clique duas vezes no ícone ODBC.

Observação No Microsoft Windows 3.0a, inicie o Administrador ODBC clicando duas vezes no ícone Administrador ODBC no grupo Microsoft ODBC.

- 2 Na caixa de diálogo **Origens de Dados**, selecione a origem de dados da lista "Origens de dados".
- 3 Escolha o botão "Excluir" e, então, escolha o botão "Sim" para confirmar a exclusão.

Ver também

Para todos os usuários

[Conectando-se a uma origem de dados dBASE](#)

[Configurando o controlador ODBC para o dBASE](#)

Origem de dados dBASE

A origem de dados dBASE especifica um diretório padrão para os dados, onde o controlador ODBC para o dBASE procura o arquivo dBASE que você deseja acessar, assim como outras informações sobre bancos de dados e controladores.

Caixa de diálogo Configuração do ODBC para o dBASE

Ver também

A caixa de diálogo **Configuração do ODBC para o dBASE** contém os seguintes campos:

Nome da origem de dados

Um nome que identifica a origem de dados, como Folha de Pagamento ou de Pessoal.

Descrição

Uma descrição opcional dos dados na origem de dados, por exemplo, "Data de admissão", "Histórico de salários" e "Atualização de empregados".

Versão

Permite selecionar a versão dos arquivos do dBASE usados: dBASE III ou dBASE IV.

Diretório

Apresenta o diretório atualmente selecionado. Antes de adicionar a origem de dados, você tem que escolher o botão "Selecionar diretório" para selecionar um diretório diferente. Ou, selecionar a caixa de verificação "Usar diretório atual" para utilizar o diretório atual de trabalho do aplicativo.

Quando você definir o diretório da origem de dados dBASE, deve especificar o diretório onde os arquivos dBASE mais usados estão alocados. O controlador ODBC para o dBASE usa este diretório como o padrão. Copie outro arquivo dBASE neste diretório, caso ele seja usado frequentemente. Como alternativa, você pode inserir nomes de arquivos com a instrução SELECT usando o nome do diretório:

```
SELECT * FROM C:\MEUDIR\EMP
```

Ou, pode utilizar a instrução USE para especificar um novo diretório padrão:

```
USE C:\MEUDIR
```

Selecionar diretório

Exibe a caixa de diálogo onde você pode selecionar o diretório contendo os arquivos dBASE a serem acessados.

Selecionar índices

Exibe a caixa de diálogo Selecionar Índices onde você pode associar arquivos dBASE com arquivos de índice.

Usar diretório atual

Quando selecionado, torna o diretório atual do aplicativo em uso no diretório da origem de dados e desativa a opção "Selecionar diretório". Quando desativado, permite a você selecionar o diretório da origem de dados usando a opção "Selecionar diretório".

Controlador

Exibe as seguintes opções:

Atenção A não ser para a opção "Exclusivo", estas opções se aplicam a todas as origens de dados que usam o controlador ODBC para o dBASE e o controlador ODBC para o Microsoft FoxPro.

Seqüência agrupada

Seqüência na qual os campos são ordenados: ASCII ou Internacional.

Tempo limite de página

Especifica o período de tempo, em décimos de segundo, que uma página (se não for usada) permanece no buffer antes de ser removida. O padrão é 600 décimos de segundo (60 segundos).

Exclusivo

Se a caixa "Exclusivo" for selecionada, o arquivo dBASE será aberto no modo Exclusivo e

poderá ser acessado apenas por um único usuário de cada vez. Se a caixa Exclusivo for desativada, os arquivos dBASE serão abertos no modo Compartilhado e poderão ser acessados por mais de um usuário, ao mesmo tempo. Um maior desempenho é atingido quando se ativa o modo Exclusivo.

Exibir linhas excluídas

Especifica se as linhas que foram marcadas como excluídas devem ou não ser exibidas.

Contagem aproximada de linhas

Determina se os dados estatísticos do tamanho de uma tabela são aproximados.

Ver também

Para todos os usuários

[Adicionando, modificando e excluindo uma origem de dados dBASE](#)

[Caixa de diálogo **Selecionar Índices**](#)

Caixa de diálogo Selecionar Índices

Ver também

A caixa de diálogo **Selecionar Índices** contém os seguintes campos:

Tabelas

Exibe uma lista de arquivos na origem de dados atualmente selecionada.

Índices

Exibe os índices designados para o arquivo atualmente selecionado na lista "Tabelas".

Listar arquivos do tipo

Exibe e permite escolher os tipos de arquivos a serem mostrados na lista "Índices".

Para associar um índice a uma tabela

- 1 Na lista "Tabelas", selecione um arquivo.
- 2 Na lista "Índices", selecione um índice.
- 3 Escolha o botão "OK", para salvar as associações tabelas/índices.

Observação Os índices do dBASE III têm que ser designados usando-se esta caixa de diálogo para que o controlador os reconheça.

Ver também

Para todos os usuários

Caixa de diálogo **Configuração do ODBC para o dBASE**

Conectando-se a uma origem de dados dBASE

Ver também

Quando você se conecta a uma origem de dados dBASE, o programa poderá pedir que insira o nome do diretório. Neste caso, deve inserir ou selecionar o diretório contendo os arquivos de banco de dados dBASE que deseja acessar.

Ver também

Para todos os usuários

[Adicionando, modificando e excluindo uma origem de dados dBASE](#)

[Usando o controlador ODBC para o dBASE](#)

Para usuários experientes

[Seqüências de conexão \(Avançado\)](#)

Usando o controlador ODBC para o dBASE

As informações a seguir podem ser úteis quando você estiver usando o controlador ODBC para o dBASE:

Índices

O usuário deve usar a caixa de diálogo **Selecionar Índices** na opção Painel de Controle ODBC (ou Administrador ODBC se estiver executando o Microsoft Windows 3.0a) para definir índices para os arquivos do dBASE III. Caso contrário, o controlador não poderá usar ou manter os índices.

Usando palavras reservadas

Não usar como identificadores as palavras reservadas listadas na gramática SQL do Apêndice C do *Microsoft ODBC SDK Programmer's Reference* (isto é, nomes de tabelas ou colunas), a menos que você coloque a palavra entre aspas (por exemplo, "Data").

Nomes de coluna e tabela

Se um nome de coluna ou tabela contiver qualquer caractere, com exceção de letras, números e sublinhados, ele deve ser delimitado. Para delimitar um nome de coluna ou de tabela, coloque-o entre aspas (" "). Caracteres com um valor ASCII maior que 127 serão convertidos para sublinhados.

Arquivos de índice NTX

O controlador ODBC para o dBASE não suporta os arquivos de índice .NTX do Clipper.

Aprimorando o tempo de resposta

O tempo de resposta em arquivos grandes podem ser aprimorados com a construção de um índice .MDX (ou .NDX) na coluna (campo) especificada na cláusula WHERE da instrução SELECT.

Índices .MDX existentes serão automaticamente aplicados para =, >, <, >=, =< e para operadores BETWEEN em uma cláusula WHERE, e também para predicados LIKE, assim como satisfazendo predicados associados.

Seqüências de conexão (Avançado)

Ver também

A seqüência de conexão do controlador ODBC para o dBASE usa as seguintes palavras-chave:

Palavra-chave	Descrição
DSN	Nome da origem de dados dBASE.
DBQ	Diretório do banco de dados dBASE.
FIL	Tipo de arquivo (DBASE3 ou DBASE4).

Por exemplo, para se conectar a origem de dados Contabilidade no diretório C:\DBASE, use a seguinte seqüência de conexão:

```
DSN=Contabilidade;DBQ=C:\DBASE;FIL=DBASE4
```

Ver também

Para todos os usuários

[Conectando-se a uma origem de dados dBASE](#)

Instruções SQL (Avançado)

Ver também

O controlador ODBC para o dBASE suporta todas as instruções SQL e cláusulas na gramática mínima do ODBC.

Para informações sobre as limitações da gramática SQL do ODBC e outras, assim como informações sobre a gramática suportada específica do controlador, consulte os seguintes tópicos:

Gramática adicional suportada pela SQL do ODBC (Avançado)

Gramática SQL do ODBC específica do controlador (Avançado)

Limitações da gramática SQL do ODBC (Avançado)

Ver também

Para usuários experientes

[Tipos de dados \(Avançado\)](#)

Gramática adicional suportada pela SQL do ODBC (Avançado)

Ver também

O controlador ODBC para o dBASE suporta plenamente as seguintes instruções SQL e cláusulas na gramática principal e estendida do ODBC:

Gramática principal e estendida	Comentários
Literal numérica aproximada	Suportada.
AVG(<i>expressão</i>), COUNT(*), MAX(<i>expressão</i>), MIN(<i>expressão</i>) e SUM(<i>expressão</i>)	Suportada. Consulte também a descrição de COUNT(<i>expressão</i>) em <u>Gramática SQL do ODBC específica do controlador</u> .
Predicado BETWEEN	Suportada.
Nomes correlacionados são plenamente suportados, incluindo os inseridos na lista de tabelas.	Por exemplo, na seguinte seqüência de caracteres, E1 é o nome correlacionado para a tabela Emp: <pre>SELECT * FROM Emp E1 WHERE E1.Sobrenome = 'Silva'</pre>
DELETE, INSERT, e UPDATE suportam nomes de caminhos com nomes de tabelas	Por exemplo: <pre>INSERT INTO R:\MEUDIR\EMP\ MINHATAB</pre>
Literal numérica exata	Suportada.
[HAVING <i>condição de localização</i>]	Suportada.
IN (<i>lista de valores</i>)	Implementado como o especificado na gramática principal do ODBC. Por exemplo: <pre>SELECT * FROM EMP WHERE Dept IN ('Vendas', 'Marketing')</pre>

Ver também

Para usuários experientes

[Gramática SQL do ODBC específica do controlador \(Avançado\)](#)

[Limitações da gramática SQL do ODBC \(Avançado\)](#)

Gramática SQL do ODBC específica do controlador (Avançado)

Ver também

O controlador ODBC para o dBASE suporta as seguintes gramáticas SQL do ODBC específicas do controlador:

Gramática SQL do ODBC específica do controlador	Comentários
Predicado BETWEEN	A sintaxe: <i>expressão1</i> BETWEEN <i>expressão2</i> AND <i>expressão3</i> retorna True somente se a <i>expressão1</i> for maior ou igual à <i>expressão2</i> e a <i>expressão1</i> for menor ou igual à <i>expressão3</i> .
COUNT(<i>expressão</i>)	Conta todos os valores não-NULL para uma expressão através de um predicado. Esta função se comporta como outras funções definidas, tais como SUM, AVG, MIN, e MAX. Por exemplo: <pre>SELECT COUNT (A+B) FROM Q</pre> conta todas as linhas em Q onde A+B não é igual a NULL.
Aritmética de data e hora	O controlador suporta a adição e a subtração de um número inteiro de uma coluna de data. O número inteiro especifica o número de dias a ser adicionado ou subtraído. O controlador também suporta a subtração de uma coluna de data de uma outra, retornando um número de dias.
Data em formato literal	O formato AAAA-MM-DD é suportado:
GROUP BY <i>lista de expressões</i>	GROUP BY suporta uma lista de expressões bem como um nome de coluna.
ORDER BY <i>lista de expressões</i>	Se a expressão é uma literal de número inteiro único, ela é interpretada como sendo o número da coluna no resultado. A ordenação é feita em uma das colunas da tabela de resultados. Nenhuma ordenação é permitida nas funções definidas, ou numa expressão que contenha uma função definida. Por exemplo, nas seguintes cláusulas a tabela é ordenada por três expressões-chave: a+b, c+d, e.

ORDER BY com GROUP BY

```
SELECT * FROM emp  
ORDER BY a+b, c+d, e
```

ORDER BY pode ser executado em qualquer expressão no GROUP BY *lista de expressões* ou em qualquer coluna no resultado.

Relações biunívocas

Uma instrução SELECT pode conter uma lista de cláusulas OUTER JOIN.

Funções escalares

Nomes de tabelas que ocorrem na cláusula FROM da SELECT, depois da cláusula INTO na INSERT e após CREATE e DROP TABLE, podem conter um nome de caminho válido, um nome principal e uma extensão do nome de arquivo.

Suportadas.

O uso de um nome de tabela em qualquer outro lugar na instrução SQL não suporta o uso de nomes de caminhos ou extensões, aceitando somente um nome principal (por exemplo, EMP FROM C:\ABC\EMP).

Se a extensão de um nome de arquivo não for especificada, a extensão .DBF será acrescentada automaticamente.

Nomes correlacionados (aliases) podem ser usados. Por exemplo:

```
SELECT * FROM  
C:\ABC\EMP T1  
WHERE T1.COL1 = 'aaa'
```

USE [*unidade de disco:*]*dir*

Define o diretório atual do banco de dados. *Unidade de disco* é um nome válido de unidade de disco e *dir* é qualquer nome válido de diretório MS-DOS.

Por exemplo, a seguinte expressão muda o diretório atual para C:\DBDIR:

```
USE C:\DBDIR
```

Ver também

Para usuários experientes

[Gramática adicional suportada pela SQL do ODBC \(Avançado\)](#)

[Limitações da gramática SQL do ODBC \(Avançado\)](#)

Relações biunívocas (Avançado)

Ver também

O controlador ODBC para o dBASE estende a gramática do OUTER JOIN para suportar as relações biunívocas aninhadas. A gramática do OUTER JOIN é:

relação biunívoca à esquerda::=

referência da tabela **LEFT OUTER JOIN** *referência da tabela*

ON *condição de localização*

referência da tabela::=

nome da tabela | [(*relação biunívoca à esquerda*)]

onde *nome da tabela* pode ser um nome de tabela ou um nome de tabela seguido de um nome correlacionado. Por exemplo, a instrução seguinte usa uma relação biunívoca tripla para criar uma lista de ordens de venda. Para cada ordem de venda todos os números de linha (se houver algum) são listados, e, para cada número de linha, o item e a descrição (se houver alguma) são listados.

```
SELECT Ordem.SONum,  
       Linha.LineNúm,  
       Part.PartNúm,  
       Part.Descrição  
FROM Ordem LEFT OUTER JOIN  
      (Linha LEFT OUTER JOIN Part  
      ON Linha.PartNúm=Part.PartNúm)  
      ON Ordem.SONúm=Linha.SONúm
```

Observação O ON da extrema direita corresponde ao LEFT OUTER JOIN da extrema esquerda.

Ver também

Para usuários experientes

[Gramática adicional suportada pela SQL do ODBC \(Avançado\)](#)

[Limitações da gramática SQL do ODBC \(Avançado\)](#)

Limitações da gramática SQL do ODBC (Avançado)

Ver também

O controlador ODBC para o dBASE impõe as seguintes limitações à gramática SQL do ODBC:

Gramática	Limitação
Seqüência de caracteres literais	Qualquer caractere ANSI (1 - 255 decimal). O apóstrofo (') é representado usando-se dois apóstrofes consecutivos (").
Colunas	Quando uma tabela é criada o número máximo é de 255.
Literais	O comprimento máximo é de 1000 caracteres.
NOT NULL	Não suportado na instrução CREATE TABLE.
Predicados	O número máximo é de 300.
Chaves de classificação	O comprimento máximo de uma chave de classificação em GROUP BY, ORDER BY, instrução SELECT DISTINCT ou relação biunívoca é de 255 bytes; o comprimento máximo de todas as chaves de classificação em uma linha de classificação é de 65,500 bytes. Se o comprimento dos dados em uma coluna for maior que 255 caracteres, a classificação se baseará nos primeiros 255 caracteres.
Predicado Comparação	Para dados SQL_BIT, somente podem

	ser feitas comparações para igualdade e desigualdade (operadores = e < >).
Predicado LIKE	Se os dados em uma coluna forem maiores que 255 caracteres, a comparação LIKE se baseará somente nos primeiros 255 caracteres.
Funções definidas	AVG, MAX, MIN, e SUM não suportam a palavra-chave DISTINCT.

Ver também

Para usuários experientes

[Gramática adicional suportada pela SQL do ODBC \(Avançado\)](#)

[Gramática SQL do ODBC específica do controlador \(Avançado\)](#)

Índices do dBASE (Avançado)

O controlador ODBC para o dBASE automaticamente abre e atualiza os arquivos de índices do dBASE IV. O usuário tem que usar a opção ODBC do Painel de Controle (ou o programa Administrador ODBC, se estiver executando o Windows 3.0a) para associar os arquivos .NDX do dBASE III com os arquivos do dBASE.

As seguintes limitações se aplicam à criação de índices dBASE:

- Todos os nomes de coluna devem ser válidos.
 - Todas as colunas devem estar na mesma ordem ascendente ou descendente.
 - Se houver uma única coluna de texto, seu comprimento deve ser menor que 100 bytes.
 - Se houver mais de uma coluna, todas elas devem ser colunas de texto, e a soma dos tamanhos das colunas deve ser menor que 100 bytes.
 - Campos lógicos ou de memorando não podem ser indexados.
 - Um índice não pode ser especificado para o grupo de campos atual, isto é, não são permitidos índices duplicados.
 - O nome do índice tem que corresponder a convenção de nomes de índices do dBASE.
- O dBASE III requer cada índice como estando em um arquivo separado, cada qual com uma extensão .NDX. No dBASE IV os índices são criados como nomes de marca, que são armazenados em um único arquivo com extensão .MDX. O arquivo com extensão .MDX tem o mesmo nome base do arquivo do banco de dados (por exemplo, EMP.MDX é o arquivo de índices para o banco de dados EMP.DBF).

Tipos de dados (Avançado)

Ver também

A tabela seguinte mostra como os tipos de dados dBASE são transformados em tipos de dados SQL do ODBC. Note que nem todos os tipos de dados SQL do ODBC são suportados.

Tipos de dados do dBASE	Tipos de dados do ODBC
Caractere	SQL_CHAR
Data	SQL_DATE
Lógico	SQL_BIT
Memorando	SQL_LONGVARCHAR
Numérico (BCD)	SQL_DOUBLE

A precisão no dBASE III permite números com expoentes de até dois dígitos e no dBASE IV números com expoentes de até três dígitos. Uma vez que os números são armazenados como texto, eles são convertidos em números. Se o número a ser convertido não couber num campo, resultados inexplicados podem ocorrer.

Embora o dBASE permita que uma precisão e uma escala sejam especificadas com um tipo de dados Numérico, isto não é suportado pelo controlador ODBC para o dBASE. O controlador ODBC para o dBASE sempre retorna uma precisão de 15 e uma escala de 0 para tipos de dados Numéricos.

Uma coluna criada com o tipo de dados Numérico usando o controlador ODBC para o dBASE transforma-se em tipo de dados SQL_DOUBLE ODBC. Desta maneira, os dados nesta coluna ficam sujeitos a arredondamento. Este comportamento não é o mesmo que o do tipo de dados Numérico no dBASE (tipo N), que é Decimal Codificada em Binário (BCD).

Observação **SQLGetTypeInfo** retorna tipos de dados SQL do ODBC. Todas as conversões no Apêndice D do *Microsoft ODBC SDK Programmer's References* são suportadas pelos tipos de dados SQL do ODBC listados anteriormente neste tópico.

Limitações

O controlador ODBC para o dBASE e o dBASE impõem as seguintes limitações aos tipos de dados:

- O tamanho máximo de uma coluna LONGVARCHAR é 65.500 bytes.
 - Na conversão de dados do dBase para tipo de dados C SQL_C_TINYINT, números de 0 a 127 são convertidos corretamente. Números de 128 a 255 são convertidos a números de -128 a -1. Números menores que 0 ou maiores que 255 não podem ser convertidos.
- Na conversão de tipo de dados C SQL_C_TINYINT para dados do dBase, números de 0 a 127 são convertidos corretamente. Números de -128 a -1 são convertidos para números de 128 a 255.

Isto acontece porque a SQL_C_TINYINT tem sinal, mas o controlador ODBC para o dBase usa inteiros de um único byte sem sinal.

Ver também

Para usuários experientes

[Instruções SQL \(Avançado\)](#)

Mensagens de erro (Avançado)

Quando um erro ocorre, o controlador ODBC para o dBASE retorna o número de erro nativo, a SQLSTATE (um código de erro do ODBC) e uma mensagem de erro.

Erro nativo

Para erros que ocorrem no nível ISAM, o controlador ODBC para o dBASE devolve o erro nativo retornado a ele pela Biblioteca de Arquivos do ODBC (isto é, o ISAM do dBASE). Para erros que são detectados pelo controlador Simba, o controlador ODBC para o dBASE retorna um erro nativo de zero.

SQLSTATE

Para erros que ocorrem na origem de dados, o controlador ODBC para o dBASE transforma o erro nativo retornado na SQLSTATE apropriada. Para erros que são detectados pelo controlador ou pelo Gerenciador do Controlador, o controlador ODBC para o dBASE, ou o Gerenciador do Controlador, gera a SQLSTATE apropriada.

Mensagem de erro

Para erros que ocorrem na origem de dados, o controlador ODBC para o dBASE devolve uma mensagem de erro retornada a ele pela Biblioteca de Arquivos do ODBC. Para erros que ocorrem no controlador ODBC para o dBASE ou no Gerenciador do Controlador, o controlador ODBC para o dBASE retorna uma mensagem de erro baseada no texto associado à SQLSTATE.

As mensagens de erro têm o seguinte formato:

[vendedor][componente ODBC][origem de dados]texto da mensagem

onde os prefixos em parênteses ([]) identificam a localização do erro. Quando o erro ocorre no Gerenciador do Controlador ou no controlador Simba, a *origem de dados* não é dada. Quando o erro ocorre na origem de dados, os prefixos *[vendedor]* e *[componente ODBC]* identificam o vendedor e o nome do componente do ODBC que recebeu o erro da origem de dados.

A seguinte tabela mostra as mensagens de erro retornadas pelo Gerenciador do Controlador, pelo controlador Simba e pelo ISAM do dBASE:

<u>Mensagem de erro</u>	<u>Localização de erro</u>
<i>[Microsoft][DLL do ODBC]texto da mensagem</i>	Gerenciador do Controlador (ODBC.DLL)
<i>[Microsoft][controlador ODBC de nível único]texto da mensagem</i>	Controlador Simba (SIMBA.DLL)
<i>[Microsoft][controlador ODBC de nível único][biblioteca de arquivos do ODBC]texto da mensagem</i>	ISAM do dBASE (XBS110.DLL)

Valores retornados SQLGetInfo (Programação)

Ver também

A tabela seguinte lista os #define da linguagem C para o argumento *flInfoType* e os valores correspondentes retornados pela **SQLGetInfo**. Esta informação pode ser recuperada passando-se a lista de #define da linguagem C para a **SQLGetInfo** no argumento *flInfoType*. Onde a SQLGetInfo retorna uma máscara de 32 bits, uma barra vertical (|) representa um OR, bit a bit. Para maiores informações sobre os valores retornados pela SQLGetInfo, consulte o *Microsoft ODBC SDK Programmer's Reference, Versão 1.0*.

<u>Valor flInfoType (#define)</u>	<u>Valor retornado</u>
SQL_ACCESSIBLE_PROCEDURES	"N"
SQL_ACCESSIBLE_TABLES	"N"
SQL_ACTIVE_CONNECTIONS	0
SQL_ACTIVE_STATEMENTS	0
SQL_CONCAT_NULL_BEHAVIOR	1
SQL_CONVERT_BIGINT	0
SQL_CONVERT_BINARY	0
SQL_CONVERT_BIT	SQL_CVT_BIT
	SQL_CVT_CHAR
	SQL_CVT_DOUBLE
	SQL_CVT_LONGVARCHAR
SQL_CONVERT_CHAR	SQL_CVT_BIT
	SQL_CVT_CHAR
	SQL_CVT_DATE
	SQL_CVT_DOUBLE
	SQL_CVT_LONGVARCHAR
SQL_CONVERT_DATE	SQL_CVT_CHAR
	SQL_CVT_DATE
SQL_CONVERT_DECIMAL	0
SQL_CONVERT_DOUBLE	SQL_CVT_BIT
	SQL_CVT_CHAR
	SQL_CVT_DOUBLE
	SQL_CVT_LONGVARCHAR
SQL_CONVERT_FLOAT	SQL_CVT_BIT
	SQL_CVT_CHAR
	SQL_CVT_DOUBLE
	SQL_CVT_LONGVARCHAR
SQL_CONVERT_FUNCTIONS	SQL_FN_CVT_CONVERT
SQL_CONVERT_INTEGER	SQL_CVT_BIT
	SQL_CVT_CHAR
	SQL_CVT_DOUBLE
	SQL_CVT_LONGVARCHAR
SQL_CONVERT_LONGVARBINARY	0
SQL_CONVERT_LONGVARCHAR	SQL_CVT_CHAR
	SQL_CVT_LONGVARCHAR
SQL_CONVERT_NUMERIC	SQL_CVT_BIT
	SQL_CVT_CHAR
	SQL_CVT_DOUBLE
	SQL_CVT_LONGVARCHAR
SQL_CONVERT_REAL	SQL_CVT_BIT
	SQL_CVT_CHAR
	SQL_CVT_DOUBLE
	SQL_CVT_LONGVARCHAR
SQL_CONVERT_SMALLINT	SQL_CVT_BIT

	SQL_CVT_CHAR
	SQL_CVT_DOUBLE
	SQL_CVT_LONGVARCHAR
SQL_CONVERT_TIME	SQL_CVT_CHAR
SQL_CONVERT_TIMESTAMP	SQL_CVT_CHAR
	SQL_CVT_DATE
SQL_CONVERT_TINYINT	SQL_CVT_BIT
	SQL_CVT_CHAR
	SQL_CVT_DOUBLE
	SQL_CVT_LONGVARCHAR
SQL_CONVERT_VARBINARY	0
SQL_CONVERT_VARCHAR	0
SQL_CORRELATION_NAME	2
SQL_CURSOR_COMMIT_BEHAVIOR	2
SQL_CURSOR_ROLLBACK_BEHAVIOR	0
SQL_DATA_SOURCE_READ_ONLY	"N" (O driver não verifica se a unidade de disco é apenas para leitura.)
SQL_DBMS_NAME	"DBASE3" ou "DBASE4"
SQL_DBMS_VER	"3.0" or "4.0"
SQL_DEFAULT_TXN_ISOLATION	0
SQL_DRIVER_NAME	"SIMBA.DLL"
SQL_DRIVER_VER	" 1.01.nnnn" (nnnn especifica a data de compilação.)
SQL_EXPRESSIONS_IN_ORDERBY	"Y"
SQL_FETCH_DIRECTION	SQL_FD_FETCH_NEXT
SQL_IDENTIFIER_CASE	4
SQL_IDENTIFIER_QUOTE_CHAR	"" (double quotation mark).
SQL_MAX_COLUMN_NAME_LEN	10
SQL_MAX_CURSOR_NAME_LEN	18
SQL_MAX_OWNER_NAME_LEN	0
SQL_MAX_PROCEDURE_NAME_LEN	0
SQL_MAX_QUALIFIER_NAME_LEN	66
SQL_MAX_TABLE_NAME_LEN	12
SQL_MULT_RESULT_SETS	"N"
SQL_MULTIPLE_ACTIVE_TXN	"N"
SQL_NON_NULLABLE_COLUMNS	0
SQL_NUMERIC_FUNCTIONS	SQL_FN_NUM_MOD
SQL_ODBC_API_CONFORMANCE	1
SQL_ODBC_SAG_CLI_CONFORMANCE	1
SQL_ODBC_SQL_CONFORMANCE	0
SQL_ODBC_SQL_OPT_IEF	"N"
SQL_OUTER_JOINS	"Y"
SQL_OWNER_TERM	""
SQL_PROCEDURE_TERM	""
SQL_PROCEDURES	"N"
SQL_QUALIFIER_NAME_SEPARATOR	"\" (barra invertida)
SQL_QUALIFIER_TERM	"DIRECTORY"
SQL_ROW_UPDATES	"Y"
SQL_SCROLL_CONCURRENCY	SQL_SCCO_READ_ONLY

SQL_SCROLL_OPTIONS	SQL_SO_FORWARD_ONLY
SQL_SEARCH_PATTERN_ESCAPE	"\" (barra invertida)
SQL_SERVER_NAME	"DBASE3" ou "DBASE4"
SQL_STRING_FUNCTIONS	SQL_FN_STR_CONCAT
	SQL_FN_STR_LCASE
	SQL_FN_STR_LEFT
	SQL_FN_STR_LENGTH
	SQL_FN_STR_LOCATE
	SQL_FN_STR_LTRIM
	SQL_FN_STR_RIGHT
	SQL_FN_STR_RTRIM
	SQL_FN_STR_SUBSTRING
	SQL_FN_STR_UCASE
SQL_SYSTEM_FUNCTIONS	SQL_FN_SYS_DBNAME
	SQL_FN_SYS_USERNAME
SQL_TABLE_TERM	"TABLE"
SQL_TIMEDATE_FUNCTIONS	SQL_FN_TD_CURDATE
	SQL_FN_TD_CURTIME
	SQL_FN_TD_DAYOFMONTH
	SQL_FN_TD_DAYOFWEEK
	SQL_FN_TD_MONTH
	SQL_FN_TD_YEAR
SQL_TXN_CAPABLE	0
SQL_TXN_ISOLATION_OPTIONS	0

Ver também

Para usuários experientes

[Instruções SQL \(Avançado\)](#)

[Tipos de dados \(Avançado\)](#)

Para programadores

[Funções escalares \(Programação\)](#)

Funções escalares (Programação)

O controlador ODBC para o dBASE suporta as seguintes funções escalares:

CONCAT	LCASE	RIGHT
CONVERT	LEFT	RTRIM
CURDATE	LENGTH	SUBSTRING
CURTIME	LOCATE	UCASE
DATABASE	LTRIM	USER
DAYOFMONTH	MOD	YEAR
DAYOFWEEK	MONTH	

Para maiores informações sobre os argumentos e valores de retorno das funções escalares, consulte o Apêndice G do *Microsoft ODBC SDK Programmer's Reference*.

Funções API do ODBC (Programação)

Ver também

O controlador ODBC para o dBASE suporta todas as funções Principais e de Nível 1 e as seguintes funções de Nível 2:

- SQLDataSources
- SQLMoreResults

Estas funções API do ODBC têm as seguintes implementações com o controlador ODBC para o dBASE:

Função	Descrição
SQLDriverConnect	As seguintes palavras-chave são suportadas na <u>seqüência de conexão</u> : DSN , DBQ , e FIL .
SQLGetConnectOption e SQLSetConnectOption	Estas funções suportam as opções de conexão SQL_ACCESS_MODE, SQL_CURRENT_QUALIFIER, SQL_OPT_TRACE, e SQL_OPT_TRACEFILE SQLGetConnectOption também suporta a opção SQL_AUTOCOMMIT.
SQLGetData	Esta função pode recuperar dados de qualquer coluna, havendo ou não colunas vinculadas posteriores e independentemente da ordem em que as colunas são recuperadas.
SQLGetInfo	A SQLGetInfo suporta informação específica do controlador tipo SQL_FILE_USAGE (65002). O valor retornado é um inteiro de 16 bits que indica como o controlador trata arquivos em uma origem de dados: 0 (SQL_FILE_NOT_SUPPORTED) = O controlador não é de nível único. 1 (SQL_FILE_TABLE) = Um controlador de nível único trata campos em origens de dados como tabelas. 3 (SQL_FILE_QUALIFIER) = Um controlador de nível único trata arquivos em origens de dados como qualificadores. O controlador ODBC para o dBase retorna 1, uma vez que cada arquivo dBase é uma tabela.
SQLGetStmtOption e SQLSetStmtOption	Estas funções suportam as opções de instruções SQL_MAX_LENGTH, SQL_MAX_ROWS e SQL_NOSCAN. Elas também suportam uma opção de instrução <u>SQL_LOCK TABLES</u> específica do controlador.
SQLGetTypeInfo	Somente os nomes de tipos de dados retornados pela SQLGetTypeInfo podem ser usados com as instruções CREATE.

SQLMoreResults

SQLMoreResults sempre retorna SQL_NO_DATA_FOUND. Não pode retornar resultados adicionais.

SQLSetCursorName e SQLGetCursorName

Estas funções são suportadas, mas não podem ser usadas para atualizações posicionadas ou excluídas (por exemplo, WHERE CURRENT OF *nome do cursor*).

SQLTables

Os nomes de tabela retornados pela **SQLTables** não têm nenhuma extensão de nome de arquivo.

SQLTransact

O controlador ODBC para o dBASE não suporta transações; todas as instruções SQL que modificam dados estão sempre subordinadas. Assim, **SQLTransact** sempre retorna SQL_SUCCESS quando *fType* é SQL_COMMIT e sempre retorna SQL_ERROR quando *fType* é SQL_ROLLBACK.

Ver também

Para usuários experientes

[Mensagens de erro \(Avançado\)](#)

Para programadores

[Tópicos de implementação \(Programação\)](#)

Opção de instrução SQL_LOCK_TABLES (Programação)

O controlador ODBC para o dBASE suporta uma opção de instrução SQL_LOCK_TABLES específica do controlador, que suporta os valores DEFLOCK e XLOCK. Quando o valor da expressão é DEFLOCK, as tabelas usadas pelo *hstmt* estão sujeitas somente aos mecanismos de proteção padrão do controlador ODBC para o dBASE. Esta é a configuração padrão.

Quando o valor da opção é XLOCK, todas as tabelas usadas pelo *hstmt* são exclusivamente protegidas quando as instruções **SELECT**, **UPDATE**, **INSERT**, **DELETE** ou **CREATE INDEX** são executadas no *hstmt*. As tabelas permanecem protegidas até que o *hstmt* seja solto (chamando **SQLFreeStmt** com a opção SQL_DROP) ou a opção seja definida para DEFLOCK e o *hstmt* seja executado novamente.

As tabelas protegidas só podem ser usadas pelo protetor *hstmt*; elas não podem ser usadas por nenhum outro *hstmt*. Por exemplo, no código a seguir, a última função chamada gera uma violação de acesso porque ela usa um *hstmt* diferente:

```
SQLSetStmtOption(hstmt1, SQL_LOCK_TABLES, XLOCK);
SQLExecDirect(hstmt1, "SELECT * FROM EMP", SQL_NTS);
SQLExecDirect(hstmt1, "UPDATE EMP SET DEPT=12", SQL_NTS);
/* Esta chamada gera uma violação de acesso */
SQLExecDirect(hstmt2, "SELECT * FROM EMP", SQL_NTS);
```

Além disso, **SQLColumns**, **SQLSpecialColumns** e **SQLStatistics** não podem recuperar informações sobre uma tabela protegida em outro *hstmt*.

Atenção Procure evitar proteger mais arquivos que o necessário. Se o mesmo *hstmt* for usado para executar instruções para muitas tabelas diferentes, todas as tabelas permanecerão protegidas até que o *hstmt* seja solto ou, protegendo, seja desativado.

A opção de instrução SQL_LOCK_TABLES usa os seguintes #define:

```
#define SQL_LOCK_TABLES 1153
#define DEFLOCK 0
#define XLOCK 1
```

A opção de instrução SQL_LOCK_TABLES não pode ser usada com **SQLSetConnectOption**.

Tópicos de implementação (Programação)

Os seguintes tópicos específicos de implementação podem afetar o uso do controlador ODBC para o dBASE.

Erros aritméticos

O Controlador ODBC para o dBASE avalia a cláusula WHERE em uma instrução SELECT à medida que ela busca cada linha. Se uma linha contém um valor que causa um erro aritmético, tal como divisão por zero ou estouro numérico, o controlador retorna todas as linhas mas retorna erros para colunas com erros aritméticos. Na inserção ou atualização, entretanto, o controlador ODBC para o dBASE pára de inserir ou atualizar dados quando o primeiro erro aritmético é encontrado.

Fechando tabelas abertas (Arquivos)

Chamar a **SQLFreeStmt** com a opção SQL_CLOSE muda o estado da instrução, mas não se fecha as tabelas usadas pelo *hstmt*. Para fechar as tabelas atualmente usadas pelo *hstmt*, o usuário tem que chamar a **SQLFreeStmt** com a opção SQL_DROP.

No exemplo seguinte, quando a **SQLFreeStmt** é chamada, as tabelas Emp e Dept permanecem abertas:

```
SQLPrepare(hStmnt, "SELECT * FROM emp, dept
WHERE emp.dept = dept.dept_id", SQL_NTS);
SQLExecute(hStmnt);
/* .SQLFetch until NO_DATA_FOUND
SQLFreeStmt(hStmnt, SQL_CLOSE);
SQLPrepare(hStmnt, "SELECT * FROM emp", SQL_NTS);
```

Observação Cada arquivo usado pelo controlador ODBC para o dBase requer um gerenciador (handle) de arquivo. Como tabelas (arquivos) permanecem abertas até que a **SQLFreeStmt** seja chamada com a opção SQL_DROP, reutilizar um *hstmt* para diferentes tabelas, sem soltá-lo, pode resultar em um erro, causado por se tentar abrir arquivos demais. Observe também que os índices dBase e os campos Memo requerem arquivos separados, assim, a abertura de uma única tabela pode utilizar diversos gerenciadores de arquivo.

Modo único usuário versus modo multiusuário

O controlador ODBC para o dBASE pode usar uma origem de dados dBASE no modo único usuário ou no modo multiusuário. O usuário especifica que modo usar na caixa de diálogo **Configuração do ODBC para o dBASE**. O controlador ODBC para o dBASE permite somente a um usuário de cada vez conectar-se a uma origem de dados de um único usuário. Quando o controlador usa um arquivo na origem de dados, como quando processa uma instrução SELECT ou UPDATE, ele bloqueia exclusivamente o arquivo.

O controlador ODBC para o dBASE permite a qualquer número de usuários conectarem-se a uma origem de dados de multiusuário, simultaneamente. Ele bloqueia exclusivamente registros antes de atualizá-los ou excluí-los. Outros usuários não podem atualizar ou excluir um registro bloqueado. Se eles podem ou não ler o registro depende da versão do arquivo. Outros usuários não podem ler registros bloqueados do dBASE III; eles podem ler registros bloqueados do dBASE IV (são permitidos "dirty reads").

Observação Embora o controlador ODBC para o dBASE bloqueie registros no modo multiusuário, isto não implica que transações em série estejam disponíveis. O controlador ODBC para o dBASE não suporta transações, assim como não suporta quaisquer níveis de isolamento de transação.

Classificando com DISTINCT, GROUP BY, JOIN, ou ORDER BY

DISTINCT, GROUP BY, JOIN, e ORDER BY sempre resultam em uma classificação. Se os índices forem encontrados, os dados serão buscados dinamicamente e a classificação será baseada no uso destes índices. Se os índices não forem encontrados, uma tabela

temporária chamada instantâneo será formada a partir dos dados e a classificação se dará na tabela temporária. Este tipo de classificação não é baseado em dados dinâmicos, uma vez que a tabela temporária é construída a partir de dados encontrados no arquivo de dados original durante a **SQLExecute**.

API

Interface de programação de aplicativo. Um conjunto de rotinas que um aplicativo como, por exemplo, o Microsoft Access utiliza para solicitar e executar serviços de nível inferior.

nível de conformidade

Alguns aplicativos podem usar somente controladores que dão suporte a certos níveis de funcionalidade ou níveis de conformidade. Por exemplo, um aplicativo pode exigir que os controladores tenham a capacidade de pedir ao usuário uma senha para uma origem de dados. Este recurso é parte do nível de conformidade 1 da interface de programação de aplicativo (API).

Cada um dos controladores ODBC está de acordo com um dos três níveis API (Central/Core, Nível 1/Level 1 ou Nível 2/Level 2) e um dos três níveis de gramática SQL (Mínimo/Minimum, Central/Core ou Ampliado/Extended). Os controladores podem dar suporte a alguns dos níveis de funcionalidade em níveis acima de seus níveis definidos.

Para obter informações detalhadas sobre níveis de conformidade, os programadores devem consultar o manual *Microsoft Open Database Connectivity Programmer's Reference*.

origem de dados

Uma origem de dados inclui os dados que um usuário deseja acessar e as informações necessárias para chegar até aqueles dados. Exemplos de origens de dados são:

- Um diretório contendo um conjunto de arquivos dBASE que você deseja acessar.
- Um banco de dados SQL Server, o servidor onde ele está gravado e a rede usada para acessar este servidor.

DBMS

Sistema de gerenciamento de banco de dados (SGBD). O software usado para organizar, analisar, localizar, atualizar e recuperar dados.

DDL

Linguagem de definição de dados. Qualquer instrução SQL pode ser usada para definir objetos de dados e seus atributos. Os exemplos incluem instruções CREATE TABLE, DROP VIEW e GRANT.

DLL

Biblioteca de vínculo dinâmico. Um conjunto de rotinas que um ou mais aplicativos podem utilizar para executar tarefas rotineiras. Os controladores ODBC são DLLs.

DML

Linguagem de manipulação de dados. Qualquer instrução SQL pode ser usada para manipular dados. Os exemplos incluem instruções UPDATE, INSERT e DELETE.

ODBC

Open Database Connectivity (conectividade de banco de dados aberto). Um gerenciador de controladores e um conjunto de controladores ODBC que permitem aos aplicativos acessar dados usando SQL como linguagem padrão.

Gerenciador de Controladores ODBC

Uma biblioteca de vínculo dinâmico (DLL) que permite o acesso a controladores ODBC.

Controlador ODBC

Uma biblioteca de vínculo dinâmico (DLL) que um aplicativo com capacidade ODBC como, por exemplo, o Microsoft Excel pode usar para ter acesso a uma determinada origem de dados. Cada sistema de gerenciamento de banco de dados (DBMS/SGBD) como, por exemplo, o SQL Server ou o dBASE, exige um controlador diferente.

SQL

Structured Query Language (linguagem de consulta estruturada). Uma linguagem usada para a recuperação, atualização e gerenciamento de dados.

Instrução SQL

Um comando escrito em linguagem de consulta estruturada (Structured Query Language - SQL); também conhecido como uma consulta. Uma instrução SQL especifica uma operação a ser executada, tal como SELECT, DELETE, ou CREATE TABLE; as tabelas e colunas onde a operação deve ser executada; e qualquer restrição àquela operação.

opção de tradução

Uma opção que especifica a maneira como um tradutor traduz os dados. Por exemplo, uma opção de tradução pode especificar os conjuntos de caracteres entre os quais o tradutor traduz dados de caracteres. Ele também pode fornecer uma chave para a codificação e decodificação.

tradutor

Uma biblioteca de vínculo dinâmico (DLL) que traduz todos os dados que passam por um aplicativo como, por exemplo, o Microsoft Access e uma origem de dados. A utilização mais comum de um tradutor é para traduzir dados de caracteres entre diferentes conjuntos de caracteres. Um tradutor também pode executar tarefas como, por exemplo, codificação e decodificação ou compressão e expansão.

