

Query Management Facility



Iniciação ao QMF para Windows

Versão 7

Query Management Facility



Iniciação ao QMF para Windows

Versão 7

Atenção!

Antes de utilizar as instruções contidas neste manual, bem como o produto a que elas se referem, não deixe de ler as informações gerais incluídas em “Apêndice. Informações” na página 145.

Quinta Edição (Setembro 2000)

Esta edição aplica-se ao Query Management Facility para Windows, uma função da Versão 7 do DB2 Universal Database Server para OS/390 (DB2 UDB para OS/390), 5675-DB2, ao Query Management Facility, uma função da Versão 7 do DATABASE 2 Server para VM e VSE (DB2 para VM e VSE), 5697-F42, ao Query Management Facility para Windows para AS/400, 5697-G24 e ao Query Management Facility para Windows para DB2 Workstation Databases, 5697-G22, do DB2 Warehouse Manager, 5648-D35, do DB2 Warehouse Manager para AS/400, 5697-G23 e a todas as edições e modificações seguintes, até indicação em contrário patente em novas edições.

Esta edição substitui e inutiliza a edição anterior, SC26-9582-02. As alterações técnicas a esta edição estão indicadas por uma barra vertical ao lado da alteração. Alterações editoriais que não tenham relevância técnica não estão marcadas.

Índice

A Biblioteca QMF	vii	Imprimir consultas SQL	15
		Ver antes uma consulta	15
		Imprimir consultas SQL	15
Capítulo 1. Introdução	1	Capítulo 3. Trabalhar com Consultas	
Servidores de Bases de Dados	1	Orientadas	17
Família de bases de dados de DB2	1	Construir consultas simples	17
Nome de utilizador vs. nome técnico	1	Abrir novas consultas orientadas	17
Definir o nome do servidor	1	Botões de acção de consulta orientada	17
Segurança da base de dados	2	Adicionar tabelas a consultas orientadas	18
Iniciar sessão	2	Executar consultas orientadas	18
Corrigir palavras-passe	3	Construir Consultas Complexas	19
Alterar palavras-passe	3	Adicionar colunas a consultas orientadas	19
Especificar cadeias de contabilização	4	Utilizar condições de ordenação	20
Regulação	4	Adicionar condições de ordenação	20
Ver limites de recurso	4	Utilizar condições de linha	20
Definir o seu limite de linha	5	Adicionar condições de linha	21
Personalizar a barra de ferramentas	6	Utilizar tabelas múltiplas em consultas orientadas	22
Adicionar botões à barra de ferramentas	6	Criar condições de junção de consulta orientada	22
Mover botões na barra de ferramentas	6	Consultas orientadas e SQL	22
Remover botões da barra de ferramentas	7	Ver SQL para consultas orientadas	23
		Converter consultas orientadas em SQL	23
Capítulo 2. Trabalhar com Consultas SQL	9	Utilizar Variáveis de Substituição em Consultas Orientadas	23
Consultas SQL	9	Guardar Consultas Orientadas	23
Criar novas consultas SQL	9	Guardar consultas orientadas em ficheiros	23
Executar consultas SQL num servidor de base de dados	9	Abrir ficheiros de consulta orientada guardados	24
Alternar entre a vista Resultados e a vista SQL	9	Guardar consultas orientadas no servidor de base de dados	24
Trabalhar com tipos de letra	10	Abrir consultas orientadas guardadas no servidor de base de dados	24
Seleccionar o tipo de letra de apresentação da consulta	10	Imprimir Consultas Orientadas	25
Consultas múltiplas	10	Pré-visualizar consultas orientadas	25
Apresentar consultas múltiplas simultaneamente	10	Capítulo 4. Trabalhar com Resultados de Consulta	27
Extrair consultas	11	Ordenar e dimensionar resultados de consulta	27
Criar novas consultas SQL	11	Seleccionar colunas e linhas	27
Variáveis de substituição em consultas SQL	12	Redimensionar colunas e linhas	27
Executar consultas SQL com variáveis de substituição	12	Ajustar automaticamente colunas e linhas	27
Guardar e Abrir consultas SQL	13	Ordenar resultados de consulta	28
Guardar consultas SQL em ficheiros	13	Reordenar colunas	28
Abrir ficheiros de consulta SQL guardados	13		
Guardar consultas SQL no servidor de base de dados	14		
Abrir consultas SQL guardadas no servidor de base de dados	14		

Formatar resultados de consulta	28	Abrir um ficheiro de procedimento guardado	42
Seleccionar o tipo de letra de apresentação dos resultados de consulta	29	Guardar um procedimento no servidor de base de dados	42
Formatar resultados de consulta numéricos	29	Abrir procedimentos guardados no servidor de base de dados	43
Converter a formatação de resultados de consulta num formulário.	29	Imprimir procedimentos	43
Agrupar e agregar resultados de consulta	29	Pré-visualizar um procedimento	43
Agrupar resultados de consulta	29	Imprimir um procedimento	44
Resumir resultados de consulta	29		
Guardar resultados de consulta e formatação	30	Capítulo 7. Trabalhar com Listas	45
Guardar resultados de consulta como uma tabela	30	Objectos	45
Guardar resultados de consulta em ficheiros	30	Listar objectos	45
Imprimir resultados de consulta	30	Comandos da janela Lista	46
Pré-visualizar resultados de consulta.	30	Criar listas	46
Imprimir resultados de consulta	30	Adicionar objectos a listas	46
		Remover objectos de listas	47
		Guardar listas em ficheiros	47
		Abrir ficheiros de lista guardados.	47
Capítulo 5. Trabalhar com Relatórios	33		
Formulários	33	Capítulo 8. Trabalhar com Ficheiros de Trabalho	49
Compreender formulários	33	Ficheiros de trabalho	49
Elaborar um relatório usando um formulário	34	Criar ficheiros de trabalho	49
Editar um formulário	34	Executar ficheiros de trabalho	49
Criar um formulário	35	Ajustar automaticamente colunas e linhas	49
Etapa 1: Criar um formulário	35	Ordenar resultados de consulta	49
Etapa 2: Alterar a ordem das colunas	35	Reordenar colunas	50
Etapa 3: Alterar os títulos de coluna	35	Formatar resultados de consulta	50
Etapa 4: Alterar o formato da coluna.	36	Seleccionar o tipo de letra de apresentação dos resultados de consulta	50
Etapa 5: Adicionar informações de resumo	36	Formatar resultados de consulta numéricos	51
Etapa 6: Adicionar cabeçalhos e pés-de-página	36	Converter a formatação de resultados de consulta num formulário.	51
Guardar formulários	37	Agrupar e agregar resultados de consulta	51
Guardar um formulário em ficheiro	37	Agrupar resultados de consulta	51
Abrir ficheiros de formulários guardados	37	Resumir resultados de consulta	51
Guardar formulários no servidor de base de dados	37	Guardar resultados de consulta e formatação	51
Abrir formulários guardados no servidor de base de dados	38	Guardar resultados como uma tabela	51
Imprimir relatórios.	38	Guardar resultados de consulta em ficheiros	52
Exportar relatórios	39	Imprimir resultados de consulta	52
		Pré-visualizar resultados de consulta.	52
		Imprimir resultados de consulta	52
Capítulo 6. Trabalhar com Procedimentos	41		
Executar procedimentos	41	Capítulo 9. Trabalhar com consultas estáticas	53
Criar um novo procedimento linear	41	Consultas estáticas	53
Criar um novo procedimento com lógica	41	Criar consultas estáticas	53
Executar um procedimento num servidor de base de dados	41		
Guardar procedimentos	42		
Guardar procedimentos em ficheiros.	42		

Substituir variáveis de substituição por variáveis de sistema central	54	BindSection().	76
Executar uma consulta estática	55	CancelBind().	77
Capítulo 10. Trabalhar com o Editor de Tabelas	57	ChangePassword().	77
Editor de tabelas	57	ClearList().	78
Procurar linhas usando o editor de tabelas	57	Close().	78
Adicionar uma linha	58	Commit().	79
Alterar uma linha	58	CompleteQuery().	79
Eliminar uma linha	58	CopyToClipboard().	80
Editar tabelas na vista de resultados de consulta	59	DeleteQMFObject().	81
Eliminar uma linha da vista de resultados de consulta	59	EndBind().	81
Actualizar colunas na vista de resultados de consulta	59	Execute().	82
Formulários de DB2	59	ExecuteEx().	82
Capítulo 11. Distribuir Dados.	61	ExecuteStoredProcedure().	83
Exportar dados	61	ExecuteStored ProcedureEx().	84
Exportar dados para ficheiros	61	Export().	86
Importar dados	62	ExportForm().	88
Guardar dados num servidor de base de dados	63	ExportReport().	88
Utilizar o comando Enviar Para	63	FastSaveData().	90
Utilizar o Suplemento para Microsoft Excel	64	FetchNextRow().	91
Utilizar Aplicações Exemplo	64	FetchNextRowEx().	92
Capítulo 12. Utilizar o Centro de Relatórios do QMF	65	FetchNextRows().	92
Iniciação ao Centro de Relatórios do QMF	65	FetchNextRowsEx().	94
Janela do Centro de Relatórios do QMF	65	FlushQMFCache().	94
Ligar ao Servidor	66	GetColumnCount().	95
Trabalhar com Relatórios e Objectos	67	GetColumnDataValue().	95
Executar Relatórios.	68	GetColumnHeader().	96
Trabalhar com Pastas e Favoritos	68	GetColumnHeaderEx().	96
Adicionar Relatórios a Favoritos	69	GetColumnHeadings().	97
Capítulo 13. Utilizar a API do QMF para Windows.	71	GetColumnValue().	98
Controlar o QMF para Windows através da API	71	GetColumnValueEx().	98
Bloquear chamadas	72	GetDefaultServerName().	99
Ligar à base de dados.	72	GetGlobalVariable().	99
Referência API	73	GetHostVariableNames().	99
AddDecimalHostVariable().	73	GetHostVariableTypes().	100
AddHostVariable().	73	GetLastErrorString().	100
BindDecimalHostVariable().	75	GetLastErrorType().	101
BindHostVariable().	75	GetLastSQLCode().	102
		GetLastSQLException().	103
		GetLastSQLState().	104
		GetOption().	104
		GetOptionEx().	106
		GetProcText().	106
		GetProcVariables().	107
		GetQMFObjectInfo().	108
		GetQMFObjectInfoEx().	109
		GetQMFObjectList().	111
		GetQMFObjectListEx().	112
		GetQMFProcText().	113
		GetQMFQueryText().	113
		GetQueryText().	114

GetQueryVerb()	114
GetResourceLimit()	115
GetResourceLimitEx()	119
GetRowCount()	120
GetServerList()	120
GetServerListEx()	121
GetStoredProcedureResultSets()	122
GetVariables()	122
GetVariablesEx()	123
InitializeProc()	124
InitializeQuery()	124
InitializeServer()	125
InitializeStaticQuery()	126
IsStatic()	127
Open()	127
Prepare()	128
PrintReport()	129
ReinitializeServer()	129
Rollback()	129
RunProc()	130
SaveData()	130
SaveQMFProc()	133

SaveQMFQuery()	133
SetBindOption()	134
SetBindOwner()	136
SetBusyWindowButton()	136
SetBusyWindowMessage()	137
SetBusyWindowMode()	137
SetBusyWindowTitle()	138
SetGlobalVariable()	139
SetHostVariable()	139
SetOption()	140
SetParent()	141
SetProcVariable()	142
SetVariable()	142
ShowBusyWindow()	143
StartBind()	143




Apêndice. Informações 145

Marcas Comerciais	148
-----------------------------	-----

Índice Remissivo 149

A Biblioteca QMF

Pode-se encomendar manuais através de um representante da IBM ou contactando 1-800-879-2755 nos Estados Unidos ou em qualquer dos seus territórios (Portugal 21 791 5000).

Avaliação	<div data-bbox="525 427 646 574"><p>Introdução ao QMF</p><p>GC26-9576</p></div>			
Instalação, planeamento, administração e diagnóstico	<div data-bbox="525 597 646 744"><p>Instalar e Gerir o QMF em OS/390</p><p>GC26-9575</p></div>	<div data-bbox="673 597 794 744"><p>Instalar e Gerir o QMF em VM/ESA</p><p>GC26-9573</p></div>	<div data-bbox="821 597 942 744"><p>Instalar e Gerir o QMF em VSE/ESA</p><p>GC26-9574</p></div>	<div data-bbox="969 597 1089 744"><p>Instalar e Gerir o QMF para Windows</p><p>GC26-9583</p></div>
	<div data-bbox="525 756 646 904"><p>Códigos e Mensagens do QMF</p><p>GC26-9580</p></div>	<div data-bbox="673 756 794 904"><p>Manual do Utilizador de Opção de Alto Rendimento do QMF para OS/390</p><p>SC26-9581</p></div>		
Utilização	<div data-bbox="525 927 646 1074"><p>Utilizar o QMF</p><p>SC26-9578</p></div>	<div data-bbox="673 927 794 1074"><p>Referência QMF</p><p>SC26-9577</p></div>	<div data-bbox="821 927 942 1074"><p>Iniciação ao QMF para Windows</p><p>SC26-9582</p></div>	
	Programação da Aplicação	<div data-bbox="525 1097 646 1244"><p>Desenvolver Aplicações do QMF</p><p>SC26-9579</p></div>		
	Bibliotecas Online	<div data-bbox="545 1270 655 1381"></div> <p>SK2T-0730 OS/390, VM, & VSE</p>	<div data-bbox="720 1270 830 1381"></div> <p>SK2T-6700 Apenas OS/390</p>	<div data-bbox="897 1270 1008 1381"></div> <p>SK2T-2067 Apenas VM</p>

Capítulo 1. Introdução

Este capítulo fornece uma descrição geral do QMF para Windows e explica algumas das tarefas básicas para começar com o QMF for Windows.

Servidores de Bases de Dados

Consultas, formulários, procedimentos e tabelas são executados e guardados num servidor de base de dados.

Família de bases de dados de DB2

O QMF para Windows pode ligar-se a uma grande variedade de bases de dados de DB2.

- DB2 UDB para OS/390, DB2 para OS/390 e DB2 para MVS
- Servidor de DB2 para VSE & VM e SQLDS
- DB2 Universal Database e DB2 Common Server
- DB2 Parallel Edition
- DataJoiner

A licença do QMF para Windows determina quais os produtos da família de DB2 que pode instalar e estabelecer ligação com a sua cópia do QMF para Windows.

Nome de utilizador vs. nome técnico

Versões e tipos diferentes de DB2 fazem referência a uma base de dados através de um nome RDB, nome da localização ou outro nome técnico.

Com o QMF para Windows o seu administrador pode atribuir um nome fácil de lembrar a um nome de base de dados, por exemplo, Base de Dados de Aquisições em vez de DB2P_01_PURCH.

O QMF para Windows faz referência a um servidor de base de dados ou a uma base de dados de DB2 como um "servidor."

Definir o nome do servidor

Antes de poder consultar uma base de dados, o QMF para Windows tem de saber onde é que a base de dados está armazenada.

1. A partir do menu **Ficheiro**, seleccione **Nova Consulta SQL**. Abre-se um novo documento de consulta SQL.

2. A partir do menu **Consulta**, seleccione **Definir Servidor**. Abre-se a caixa de diálogo Definir Servidor.



3. A partir da lista de servidores disponíveis, seleccione o que pretende consultar e clique em **OK**. Quando iniciar a sessão seguinte do QMF para Windows, este volta a estabelecer ligação automaticamente ao mesmo servidor.

Segurança da base de dados

Tem de providenciar um ID de utilizador e uma palavra-passe antes de poder estabelecer ligação a um servidor.

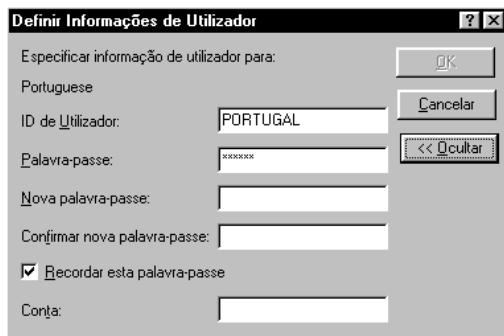
Iniciar sessão

Tem de especificar um ID de utilizador e uma palavra-passe válidos para o servidor de base de dados a que está a tentar aceder. O ID de utilizador e a palavra-passe do servidor da base de dados não são necessariamente os mesmos que o ID de utilizador e a palavra-passe locais ou de rede.

Se estiver em Windows, poderá optar por recordar palavras-passe de servidor ao longo de sessões do QMF para Windows. Se iniciou sessão actualmente em Windows, a caixa de diálogo Definir Informações de Utilizador apresenta uma caixa de verificação adicional denominada **Lembrar esta palavra-passe?**. Se seleccionar esta caixa de verificação, a palavra-passe que introduzir para esse servidor é armazenada na lista de palavras-passe do Windows. Sempre que iniciar sessão em Windows, o QMF para Windows pode recuperar automaticamente essa palavra-passe para que a mesma não lhe seja pedida. Se não tiver iniciado sessão quando executar o QMF para Windows, ou se tiver iniciado sessão como outro utilizador, o QMF para Windows pede-lhe um ID de utilizador e uma palavra-passe.

Nota: Se optar por guardar uma palavra-passe, qualquer pessoa que possa iniciar sessão na sua conta de Windows pode aceder aos seus servidores de base de dados com o seu ID de utilizador e palavra-passe (servidor).

1. A partir do menu **Consulta**, seleccione **Definir Informações de Utilizador**. Abre-se a caixa de diálogo Definir Informações de Utilizador.



2. Introduza o seu ID de utilizador e palavra-passe nos campos apropriados.

Nota: O ID de utilizador e a palavra-passe são sensíveis a maiúsculas e minúsculas. Por exemplo, se o ID de utilizador e a palavra passe estiverem em maiúsculas, tem de os introduzir em maiúsculas. Alguns tipos de servidores de bases de dados assumem que os IDs e palavras-passe são sensíveis a maiúsculas e minúsculas; outros não.

3. Marque **Lembrar esta palavra-passe?** se pretende guardar o ID de utilizador e a palavra-passe.
4. Clique em **OK**. O QMF para Windows armazena esta informação enquanto se prepara para aceder ao servidor.

Corrigir palavras-passe

Se introduziu uma palavra-passe incorrecta, pode corrigir o erro reabrindo a caixa de diálogo Definir Informações do Utilizador.

1. A partir do menu **Consulta**, clique em **Definir Informações de Utilizador**. Abre-se a caixa de diálogo Definir Informações de Utilizador.
2. Escreva a palavra-passe de novo e clique em **OK**. A palavra-passe é corrigida.

Alterar palavras-passe

Pode alterar a sua palavra-passe no servidor da base de dados a partir do QMF para Windows. Esta função é suportada actualmente apenas pelo DB2 para OS/390 Versão 5 e superior.

1. A partir do menu **Consulta**, seleccione **Definir Informações de Utilizador**. Abre-se a caixa de diálogo Definir Informações de Utilizador.

2. Clique em **Alterar**. Surgem os campos **Nova palavra-passe** e **Confirmar nova palavra-passe**.
3. Escreva a nova palavra-passe nos campos **Nova palavra-passe** e **Confirmar nova palavra-passe** e clique em **OK**. A palavra-passe do servidor da base de dados está alterada.

Especificar cadeias de contabilização

O servidor da base de dados utiliza cadeias de contabilização para rastrear a utilização do sistema. Peça ao administrador da base de dados para indicar se o sistema utiliza cadeias de contabilização.

1. A partir do menu **Consulta**, seleccione **Definir Informações de Utilizador**. Abre-se a caixa de diálogo Definir Informações de Utilizador.
2. No campo **Cadeia de contabilização**, insira a cadeia de contabilização que pretende utilizar e clique em **OK**. O QMF para Windows armazena a informação enquanto se prepara para aceder ao servidor.

Regulação

O regulador do QMF para Windows está sempre em execução no segundo plano, a supervisionar a utilização de recursos de bases de dados e de sistemas. O regulador também limita o tipo e tamanho de consultas que pode executar.

Ver limites de recurso

A partir do menu **Ver**, seleccione **Limites de Recurso**. Abre-se a caixa de diálogo Limites de Recurso. Toda a informação que consta na caixa de diálogo Limites de Recurso é só de leitura. O administrador do sistema define estes

limites.



Os tipos de limites e controlos que podem estar em efeito são:

- Tempos de espera
- Limites
- Verbos SQL
- Opções
- Guardar Dados
- Associar
- Rastreio de Objectos

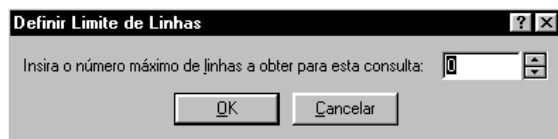
Definir o seu limite de linha

Pode especificar o número máximo de linhas a recuperar para esta consulta. Quando este limite for alcançado, o QMF para Windows cancela a consulta. O limite máximo de linhas autorizado e especificado no grupo de limites de recurso tem precedência sobre este parâmetro.

Introduza 0 para não especificar limite nenhum neste campo.

As linhas que o QMF para Windows já tenha obtido além deste limite estão retidas e disponíveis para visualização.

1. No menu **Consulta** clique em **Definir Limite de Linha**. Abre-se a caixa de diálogo Definir Limite de Linha.



2. Introduza o limite máximo de linhas que pretende que a consulta devolva e clique em **OK**. O limite de linhas será aplicado da próxima vez que executar a consulta.

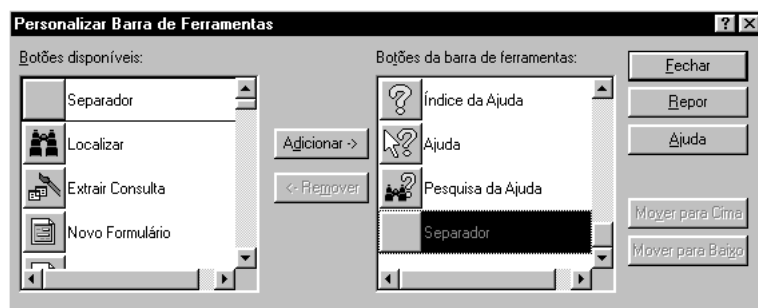
Personalizar a barra de ferramentas

Pode-se personalizar a barra de ferramentas para apresentar somente os botões que pretende ver.

Adicionar botões à barra de ferramentas

Existe a opção de adicionar botões à barra de ferramentas existente do QMF para Windows. Estes botões representam funções de que nem todos os utilizadores podem precisar, mas que estão disponíveis para inclusão na barra de ferramentas.

1. Faça duplo clique na área cinzenta em redor da barra de ferramentas. Abre-se a caixa de diálogo Personalizar Barra de ferramentas.



2. Na coluna **Botões disponíveis** seleccione o botão que pretende adicionar e clique em **Adicionar**. O botão fica na barra de ferramentas.
3. Quando acabar de adicionar botões, clique em **Fechar**. A caixa de diálogo fecha-se e os novos botões ficam na barra de ferramentas.

Mover botões na barra de ferramentas

Pode-se mudar a disposição dos botões na barra de ferramentas do QMF para Windows.

1. Faça duplo clique na área cinzenta em redor da barra de ferramentas. Abre-se a caixa de diálogo Personalizar Barra de ferramentas.
2. Na coluna **Botões disponíveis** seleccione o botão que pretende mover.
3. Use os botões **Para Cima** e **Para Baixo** para mover o botão dentro da barra de ferramentas.
4. Quando acabar de mover botões, clique em **Fechar**. A caixa de diálogo fecha-se e os botões aparecem nas novas posições.

Remover botões da barra de ferramentas

Pode-se remover botões da barra de ferramentas do QMF para Windows.

1. Faça duplo clique na área cinzenta em redor da barra de ferramentas. Abre-se a caixa de diálogo Personalizar Barra de ferramentas.
2. Na coluna **Botões disponíveis** seleccione o botão que pretende remover e clique em **Remover**. O botão será removido da barra de ferramentas.
3. Quando acabar de remover botões, clique em **Fechar**. A caixa de diálogo fecha-se e os botões já não constam da barra de ferramentas.

Capítulo 2. Trabalhar com Consultas SQL

A Structured Query Language (SQL) é a interface mais básica entre um utilizador e uma base de dados. As consultas são escritas em SQL e processadas pela base de dados. Os utilizadores podem escrever consultas do QMF para Windows em SQL ou criar consultas com o método "apontar e clicar".

Consultas SQL

As consultas de Structured Query Language exigem conhecimento dos comandos e da sintaxe SQL. Recomenda-se aos utilizadores que não estejam familiarizados com SQL que tentem criar consultas orientadas.

Criar novas consultas SQL

Clique no botão **Nova Consulta SQL** na barra de ferramentas.



Abre-se um novo documento de consulta.

Executar consultas SQL num servidor de base de dados

1. Abra um novo documento de consulta e escreva uma nova, abra um ficheiro de consulta existente ou abra uma consulta a partir da base de dados.
2. Clique no botão **Executar Consulta** na barra de ferramentas.



A consulta é executada e os resultados são apresentados.

Alternar entre a vista Resultados e a vista SQL

Pode-se ver os resultados de uma consulta ou a própria instrução SQL.

Na vista SQL de uma consulta que tenha sido executada, clique no botão **Ver Resultados** na barra de ferramentas.



São apresentados os resultados da consulta.

-ou-

Na vista Resultados de uma consulta, clique no botão **Ver SQL**.



Será apresentada a instrução SQL.

Trabalhar com tipos de letra

Pode-se alterar o tipo de letra utilizado para apresentar consultas. A escolha de tipos de letra varia consoante os elementos instalados no computador. Para mais informações sobre a adição de tipos de letra, consulte a função da ajuda do sistema operativo.

Nota: Se guardar a consulta depois de seleccionar um novo tipo de letra para apresentação da consulta, esta será sempre apresentada usando o novo tipo de letra.

Seleccionar o tipo de letra de apresentação da consulta

1. Na vista SQL, clique em **Definir Tipo de Letra** no menu **Consulta**. Abre-se a caixa de diálogo Tipos de Letra.
2. Seccione o tipo de letra para apresentação do texto da consulta e clique em **OK**. A consulta surgirá com o novo tipo de letra.

Nota: Clique em **Predefinir** para utilizar o tipo de letra seleccionado como predefinição para todas as novas consultas.

Consultas múltiplas

Pode-se ter mais do que um documento de consulta aberto em simultâneo. Também se pode executar mais do que uma consulta ao mesmo tempo. Pode-se usar esta função para gerar relatórios múltiplos ou para cortar e colar texto SQL de uma consulta para outra.

Apresentar consultas múltiplas simultaneamente

1. Abra pelo menos dois documentos de consulta.
2. No menu **Janela** seleccione um dos seguintes comandos:

Comando	Resultado
Cascata	Apresenta consultas numa série de etapas.
Mosaico Horizontal	Apresenta janelas de consulta verticalmente dispostas.

As janelas de consulta são dispostas segundo a opção que seleccionar.

Extrair consultas

Use o comando Extrair Consulta para criar novos documentos de consulta SQL. Pode especificar um ou mais nomes de tabela e o tipo de instrução SQL pretendidos e o QMF para Windows cria automaticamente uma instrução SQL que remete para os nomes e tipos de dados das colunas na dita tabela.

Criar novas consultas SQL

1. No menu **Ficheiro** clique em **Extrair Consulta**. Abre-se a caixa de diálogo Extrair Consulta.

2. Seccione o tipo de consulta que pretende criar:

Tipo de consulta	Resultado
Seleccionar	Obter linhas de uma ou mais tabelas.
Actualizar	Alterar as informações numa tabela.
Inserir	Adicionar novas linhas a uma tabela.

3. Introduza o nome e o proprietário da tabela a consultar.

Nota: Pode-se usar padrões para seleccionar nomes de tabela de uma lista de tabelas correspondentes.

- Use o carácter de percentagem (%) para fazer corresponder uma cadeia de qualquer comprimento contendo quaisquer caracteres. Por exemplo, para listar todas as tabelas com um nome começado pela letra A, introduza A%.
- Use o carácter de sublinhado (_) para fazer corresponder um único carácter. Por exemplo, para listar todas as tabelas com um proprietário que tenha a letra A na segunda posição, introduza _A%.

Depois de inserir o padrão, clique em **Adicionar da Lista** e seleccione uma tabela da lista resultante.

4. Introduza um identificador único para a tabela.
5. Clique em **Adicionar**. A tabela será adicionada à consulta.
6. Quando terminar de adicionar a(s) tabela(s) que pretende consultar, clique em **OK**. Será criada e apresentada uma consulta SQL para as tabelas seleccionadas.

Variáveis de substituição em consultas SQL

As variáveis de substituição permitem usar a mesma consulta para obter informações diferentes ao fornecer valores distintos de cada vez que se executar a consulta. Nesse sentido, para obter um conjunto de dados diferente, não é necessário reescrever a consulta. Em vez disso, basta fornecer valores diferentes para as variáveis de substituição na consulta quando a executar.

Uma variável de substituição consiste em texto que se inclui numa consulta. Este começa pelo carácter do e comercial (&) e pode conter um máximo de 18 caracteres, alfabéticos, numéricos ou um dos seguintes caracteres especiais: ^ ! \$ ~ { } ? @ # % \ ou _ . Por exemplo, seguem-se valores que constituem variáveis de substituição válidas:

```
&VARIABLE1
&DEPARTMENT_NUMBER
```

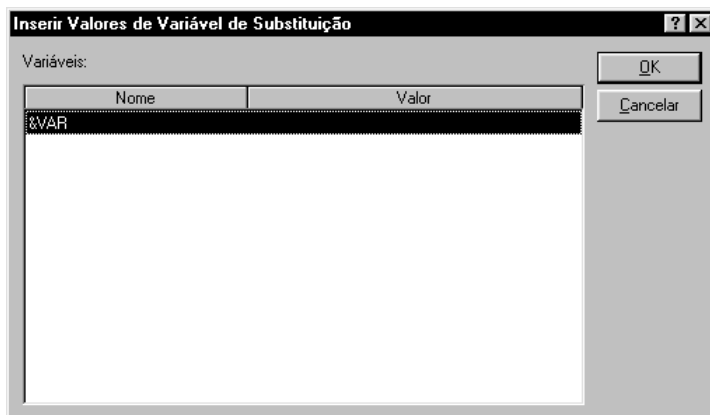
Uma variável de substituição pode aparecer algures numa consulta e o seu valor pode consistir em qualquer coisa que se escreva numa consulta (excepto um comentário). Por exemplo, pode-se usar uma variável de substituição em vez de um nome de coluna, uma condição de pesquisa, uma subconsulta ou qualquer valor específico.

Executar consultas SQL com variáveis de substituição

1. Abra um novo documento de consulta e introduza esta instrução SQL:

```
SELECT * FROM Q.STAFF WHERE DEPT >= &MIN_DEPT
```

2. Execute a consulta. Abre-se a caixa de diálogo Inserir Valores de Variável de Substituição.



3. No campo **Valor** introduza um valor de 50 e clique em **OK**. A consulta é executada e apresenta os respectivos resultados.

Tente experimentar variáveis de substituição substituindo valores nas cláusulas **SELECT** e **FROM**. Observe os resultados devolvidos pelas consultas com inputs diferentes.

Guardar e Abrir consultas SQL

Pode-se guardar consultas no PC, num servidor de ficheiros ou de base de dados.

Guardar consultas SQL em ficheiros

1. Numa consulta aberta, clique no botão **Guardar** da barra de ferramentas.



Mesmo que a consulta já tenha sido guardada, será novamente guardada. Caso contrário, abre-se a caixa de diálogo Guardar Como.

2. Introduza o nome do ficheiro em que pretende armazenar a consulta e clique em **OK**. A consulta será guardada.

Abrir ficheiros de consulta SQL guardados

1. Clique no botão **Abrir** na barra de ferramentas.



Abre-se a caixa de diálogo Abrir.

2. Selecciono o ficheiro que pretende abrir e clique em **OK**. A consulta seleccionada abre-se num novo documento de consulta.

Guardar consultas SQL no servidor de base de dados

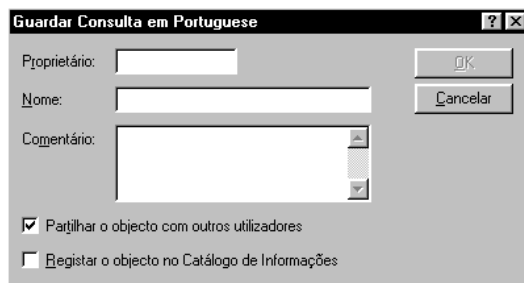
Pode-se disponibilizar o acesso a consultas guardadas no servidor a outros utilizadores. Se preferir partilhar consultas com outros utilizadores, guarde-as no servidor de base de dados.

- 1.

Numa consulta aberta, clique no botão **Guardar no Servidor** na barra de ferramentas.



Abre-se a caixa de diálogo Guardar Consulta.



2. Introduza um proprietário, um nome, seleccione se pretende ou não partilhar a consulta guardada com outros utilizadores e clique em **OK**. A consulta será guardada no servidor.
Se já existir uma consulta com este nome, ser-lhe-á pedido para sobrepor a consulta existente.

Abrir consultas SQL guardadas no servidor de base de dados

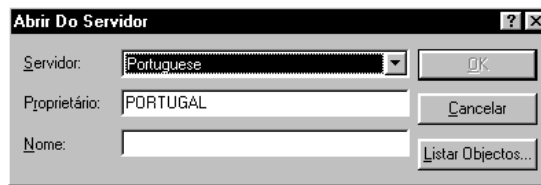
Pode-se abrir consultas que tenham sido guardadas no servidor de base de dados.

- 1.

Clique no botão **Abrir Do Servidor** na barra de ferramentas.



Abre-se a caixa de diálogo Abrir Do Servidor.



2. Introduza um servidor, proprietário e nome e clique em **OK**. Abre-se a consulta SQL.

Imprimir consultas SQL

Pode-se pré-visualizar e imprimir as consultas SQL.

Ver antes uma consulta

1. Abra uma consulta e active a vista SQL. Surge a instrução SQL.
2. No menu **Ficheiro** clique em **Configurar Página**. Abre-se a caixa de diálogo Configurar Página.
3. Proceda às alterações pretendidas ao esquema da página e clique em **OK**.
4. Clique no botão **Ver Antes** na barra de ferramentas.



Surge uma pré-visualização da consulta impressa.

Imprimir consultas SQL

1. Abra uma consulta e active a vista SQL. Surge a instrução SQL.
2. No menu **Ficheiro** clique em **Configurar Página**. Abre-se a caixa de diálogo Configurar Página.
3. Proceda às alterações pretendidas ao esquema da página e clique em **OK**.
4. Clique no botão **Imprimir** na barra de ferramentas.



A consulta será impressa.

Capítulo 3. Trabalhar com Consultas Orientadas

As consultas orientadas são uma maneira fácil de criar uma consulta seleccionando opções em menus e listas. Uma vez criada a consulta orientada, pode guardá-la ou convertê-la numa consulta SQL.

Construir consultas simples

Pode-se criar facilmente consultas simples usando a interface de consulta orientada.




Abrir novas consultas orientadas

- No menu **Ficheiro** clique em **Nova Consulta Orientada**. Abre-se um novo documento de consulta orientada.



Botões de acção de consulta orientada

Pode-se editar consultas orientadas com os botões de acção de consulta. Aparece um conjunto de botões por cima da secção controlada por este.

Botão de acção consulta orientada	Aparência	Resultado
Adicionar		Clique para adicionar um item à consulta orientada.
Editar		Clique para editar o item destacado na consulta.
Eliminar		Clique para eliminar o item seleccionado.

Para Cima e Para
Baixo



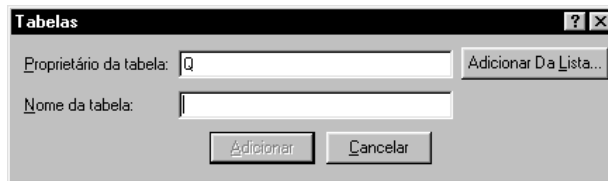
Clique para mover o item seleccionado para cima e para baixo na consulta orientada.

Adicionar tabelas a consultas orientadas

1. Na secção Tabelas do documento de consulta orientada, clique no botão **Adicionar**.



Abre-se a caixa de diálogo Tabelas.



- 2.

Escreva o nome e o proprietário da tabela que pretende adicionar e clique em **Adicionar**. A tabela será adicionada à consulta.

Nota: Pode-se usar padrões para seleccionar objectos de uma lista de objectos correspondentes.

- Use o carácter de percentagem (%) para fazer corresponder uma cadeia de qualquer comprimento contendo quaisquer caracteres. Por exemplo, para listar todas as tabelas com um nome começado pela letra A, introduza A%.
- Use o carácter de sublinhado (_) para fazer corresponder um único carácter. Por exemplo, para listar todas as tabelas com um proprietário que tenha a letra A na segunda posição, introduza _A%.

Depois de inserir o padrão, clique em **Adicionar da Lista** e seleccione uma tabela da lista resultante.

3. Adicione quaisquer outras condições de tabela à consulta e clique em **Fechar**. O documento de consulta orientada aparece com as novas tabelas listadas.

Executar consultas orientadas

Uma consulta orientada executa-se da mesma forma que uma consulta SQL. Clique no botão Executar Consulta na barra de ferramentas.



A consulta orientada é executada.

Construir Consultas Complexas

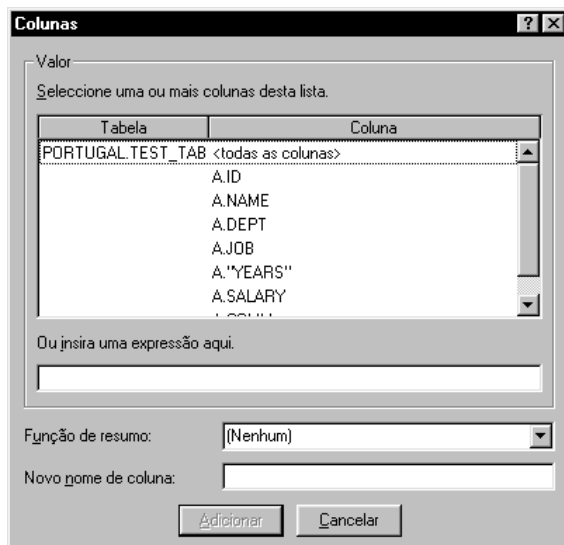
Também se pode construir consultas mais complexas usando a interface de consulta orientada.

Adicionar colunas a consultas orientadas

1. Na secção Colunas do documento de Consulta Orientada, clique no botão **Adicionar**.



Abre-se a caixa de diálogo Colunas.



2. Selecione a coluna que pretende adicionar e clique em **Adicionar**. A coluna será adicionada à consulta orientada.
3. Adicione quaisquer outras colunas à consulta e clique em **Fechar**. O documento de Consulta Orientada aparece com as novas colunas listadas.

Nota: Pode-se aplicar uma função de resumo à coluna seleccionando uma no campo Função. As funções de resumo disponíveis incluem: AVERAGE, COUNT, MAXIMUM, MINIMUM e SUM.

Nota: Pode-se mudar o nome de uma consulta escrevendo um novo nome de coluna no campo **Novo nome de coluna**.

Utilizar condições de ordenação

As condições de ordenação utilizam-se para especificar a forma como pretende ordenar as linhas na consulta, as quais podem ser dispostas por ordem ascendente (A-Z) ou descendente (Z-A).

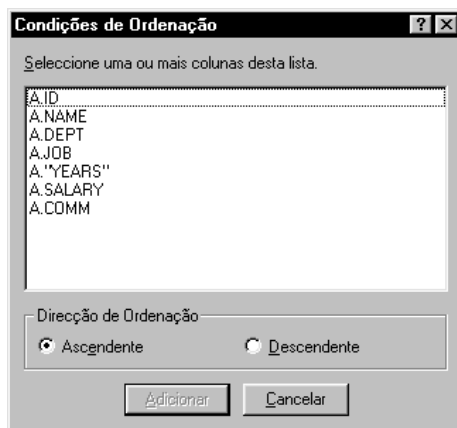
Se ordenar as linhas em mais do que uma coluna, a primeira coluna será ordenada primeiro, a segunda coluna será ordenada segundo a ordem da primeira e assim sucessivamente.

Adicionar condições de ordenação

1. Na secção Condições de ordenação do documento de Consulta Orientada, clique no botão **Adicionar**.



Abre-se a caixa de diálogo Condições de Ordenação.



2. Seleccione a coluna que pretende ordenar, a direcção de ordenação e clique em **Adicionar**. A condição de ordenação será adicionada à consulta orientada.
3. Adicione quaisquer outras condições de ordenação à consulta e clique em **Fechar**. O documento de Consulta Orientada aparece com as novas condições de ordenação listadas.

Utilizar condições de linha

Caso prefira ver somente certas linhas de uma tabela, adicione condições de linha para as seleccionar. Se não o fizer, serão apresentadas todas as linhas da tabela.

Estão disponíveis as seguintes condições de linha:

- Igual a

- Menor que
- Menor que ou igual a
- Maior que
- Maior que ou igual a
- Entre
- Começando por
- Terminando por
- Contendo
- NULL

As condições de linha são controladas pelos seguintes operadores:

- Sim É
- Não É

Adicionar condições de linha

1. Na secção Condições de linha do documento de Consulta Orientada, clique no botão **Adicionar**.



Abre-se a caixa de diálogo Condições de Linha.

Coluna	Tipo Dados
A.ID	NUMERIC
A.NAME	CHARACTER
A.DEPT	NUMERIC
A.JOB	CHARACTER
A."YEARS"	NUMERIC
A.SALARY	NUMERIC
A.COMM	NUMERIC

2. Selecciona as partes da instrução condicional e clique em **Adicionar**.

Parte da condição de linha	Função
Lado esquerdo	Selecciona a coluna que pretende examinar.
Operador	Determina a relação entre os lados esquerdo e direito da linha.

Lado direito

Introduz a condição que pretende seleccionar.

A condição de linha será adicionada à consulta orientada.

3. Adicione quaisquer outras condições de linha à consulta e clique em **Fechar**. O documento de Consulta Orientada aparece com as novas condições de linha listadas.

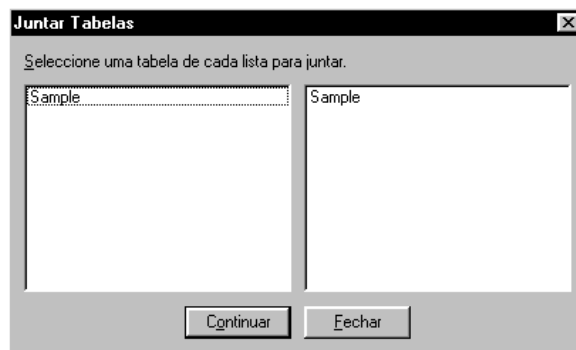
Utilizar tabelas múltiplas em consultas orientadas

Pode-se incluir informações de mais do que uma tabela numa consulta orientada.

Tem de se relacionar as duas tabelas, especificando uma ou mais condições de junção em cada uma. Só serão incluídas nos resultados linhas das tabelas em que as colunas de junção sejam iguais. O tipo de dados de cada coluna numa condição de junção deve corresponder. Uma vez especificada a relação entre duas colunas, o QMF para Windows recorda a relação e sugere-a em futuras consultas, tornando assim mais simples e eficiente a criação de posteriores consultas.

Criar condições de junção de consulta orientada

1. Na secção Tabelas do documento de Consulta Orientada, clique no botão **Adicionar** para adicionar pelo menos duas tabelas. Se nunca tiver juntado as tabelas, abrir-se-á a caixa de diálogo Juntar Tabelas. Caso contrário, o QMF para Windows sugere a condição de junção utilizada anteriormente.



2. Selecione uma coluna com o mesmo tipo de dados em cada tabela e clique em **Adicionar**. A nova condição de junção aparece na consulta orientada.

Consultas orientadas e SQL

Pode-se utilizar a interface de consulta orientada para aprender SQL.

Ver SQL para consultas orientadas

Na vista da consulta orientada, clique no botão **Ver SQL** na barra de ferramentas.



Surge a instrução SQL equivalente à consulta orientada. Não é possível modificar a instrução SQL nesta vista.

Converter consultas orientadas em SQL

Pode-se converter uma consulta orientada num novo documento de consulta SQL. A nova consulta SQL pode ser modificada, guardada, impressa e executada. No menu **Consulta** clique em **Converter em SQL**. A consulta será convertida num novo documento de consulta SQL.

Utilizar Variáveis de Substituição em Consultas Orientadas

Pode-se utilizar variáveis de substituição numa consulta orientada da mesma forma que numa consulta SQL. Consulte "Variáveis de Substituição em Consultas SQL" .

Por exemplo, as variáveis de substituição podem ser usadas em:

- uma condição de linha
DEPT É Maior Que Ou Igual A &MinDept
- uma especificação de coluna
&InputNum

Guardar Consultas Orientadas

Pode-se guardar consultas orientadas em ficheiros no PC, num servidor de ficheiros ou de base de dados.

Guardar consultas orientadas em ficheiros

1. Numa consulta orientada aberta, clique no botão **Guardar** da barra de ferramentas.



- Nota:** Mesmo que a consulta já tenha sido guardada, será novamente guardada. Caso contrário, abre-se a caixa de diálogo Guardar Como.
2. Introduza o nome do ficheiro em que pretende armazenar a consulta orientada e clique em **OK**. A consulta será guardada.

Abrir ficheiros de consulta orientada guardados

1. Clique no botão **Abrir** na barra de ferramentas.



Abre-se a caixa de diálogo Abrir.

2. Selecciono o ficheiro que pretende abrir e clique em **OK**. A consulta orientada seleccionada abre-se num novo documento de consulta.

Guardar consultas orientadas no servidor de base de dados

1. Numa consulta orientada aberta, clique no botão **Guardar no Servidor** na barra de ferramentas.



Abre-se a caixa de diálogo Guardar Consulta.

2. Introduza um proprietário, um nome, seleccione se pretende ou não partilhar a consulta guardada com outros utilizadores e clique em **OK**. A consulta será guardada no servidor.

Se já existir uma consulta com este nome, ser-lhe-á pedido para sobrepor a consulta existente.

Abrir consultas orientadas guardadas no servidor de base de dados

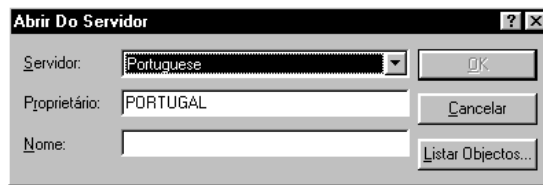
Pode-se abrir consultas orientadas que tenham sido guardadas no servidor de base de dados.

- 1.

Clique no botão **Abrir Do Servidor** na barra de ferramentas.



Abre-se a caixa de diálogo Abrir Do Servidor.



2. Introduza um servidor, proprietário e nome e clique em **OK**. Abre-se a consulta orientada.

Imprimir Consultas Orientadas

Pode-se imprimir uma consulta orientada, assim como o respectivo texto SQL. Consulte “Imprimir consultas SQL” na página 15.

Pré-visualizar consultas orientadas

Pode-se pré-visualizar os resultados ou o texto de uma consulta orientada antes de imprimir.

1. Abra uma consulta e active a vista orientada. Surge a consulta.
2. No menu **Ficheiro** clique em **Configurar Página**. Abre-se a caixa de diálogo Configurar Página.
3. Proceda às alterações pretendidas ao esquema da página e clique em **OK**.
4. Clique no botão **Ver Antes** na barra de ferramentas.



Surge uma pré-visualização da consulta impressa.

Capítulo 4. Trabalhar com Resultados de Consulta

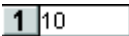



Pode-se aplicar formatação, agrupamento e agregação directamente a resultados de consulta. Esta formatação pode ser guardada com a consulta ou exportada como um formulário.

Ordenar e dimensionar resultados de consulta

Os utilizadores podem seleccionar, redimensionar, reordenar e ordenar os resultados de dados de uma consulta.

Seleccionar colunas e linhas

Uma vez a consulta executada, pode-se usar controlos na vista Resultados para editar e seleccionar informações.

Selectores de coluna e de linha	Aparência	Função
Selector de linha		Clique para seleccionar todos os dados de uma linha.
Selector de coluna		Clique para seleccionar todos os dados de uma coluna.
Célula		Clique directamente na célula para a seleccionar.
Botões Deslocar-para-final e Deslocar-para-topo		Clique para se deslocar para o topo ou final de um conjunto de resultados de consulta.

Redimensionar colunas e linhas

Pode-se alterar a aparência de um conjunto de resultados de consulta redimensionando as respectivas colunas e linhas.

1. Selecciona, com o rato, a linha divisória negra entre duas colunas ou duas linhas.
2. Arraste a linha divisória de um lado a outro ou para cima e para baixo para redimensionar a coluna ou a linha.

Nota: Se guardar a consulta depois de redimensionar linhas e colunas, a consulta será sempre apresentada com a nova formatação.

Ajustar automaticamente colunas e linhas

Pode-se dimensionar automaticamente colunas e linhas para se ajustarem aos dados que contêm.

Selecione, com o rato, uma coluna ou linha inteira e clique duas vezes na linha divisória negra entre ela e o objecto adjacente. A coluna ou a linha será automaticamente redimensionada para se ajustar aos dados.

Nota: Se guardar a consulta depois de redimensionar linhas e colunas, a consulta será sempre apresentada com a nova formatação.

Ordenar resultados de consulta

Uma vez a consulta executada, pode-se dispor os resultados em ordem alfabética por coluna.

Na vista Resultados de uma consulta, seleccione uma coluna e seleccione **Ordenação Ascendente** no menu **Resultados**.

Os resultados da consulta são dispostos por ordem ascendente.

-ou-

Na vista Resultados de uma consulta, seleccione uma coluna e seleccione **Ordenação Descendente** no menu **Resultados**.

Os resultados da consulta são dispostos por ordem descendente.

Nota: Para aplicar uma ordenação mais complexa à coluna seleccionada, seleccione **Ordenar** no menu **Resultados**.

Reordenar colunas

Pode-se alterar a ordem das colunas em resultados de consulta.

Na vista Resultados de uma consulta, seleccione uma coluna e arraste-a para a nova posição.

A coluna aparece na nova ordenação.

Formatar resultados de consulta

Pode-se alterar o tipo de letra utilizado para apresentar consultas e respectivos resultados. A escolha de tipos de letra varia consoante os elementos instalados no computador. Para mais informações sobre a adição de tipos de letra, consulte a função da ajuda do sistema operativo.

Nota: Se guardar a consulta depois de seleccionar um novo tipo de letra para apresentação dos resultados, estes serão sempre apresentados usando o novo tipo de letra.

Seleccionar o tipo de letra de apresentação dos resultados de consulta

1. Na vista Resultados, seleccione **Definir Tipo de Letra** no menu **Resultados**. Abre-se a caixa de diálogo Tipos de Letra.
2. Seleccione o tipo e o tamanho de letra para apresentação dos resultados da consulta e clique em **OK**. Os resultados da consulta serão apresentados no formato que especificou.

Nota: Clique em **Predefinir** para utilizar o tipo de letra seleccionado como predefinição para todos os resultados de consultas.

Formatar resultados de consulta numéricos

1. Na vista Resultados, seleccione uma coluna contendo valores numéricos e seleccione **Formatar** no menu **Resultados**. Abre-se a caixa de diálogo Formatar.
2. Especifique a formatação que pretende aplicar e clique em **OK**. Os valores são formatados segundo a sua selecção.

Nota: Clique em **Predefinir** para utilizar o tipo de letra seleccionado como predefinição para todos os resultados de consultas.

Converter a formatação de resultados de consulta num formulário

Pode-se converter a formatação de resultados de consulta num formulário.

1. No menu **Resultados** seleccione **Apresentar Relatório**.
Abre-se a caixa de diálogo Seleccionar Formulário.
2. Seleccione **Da Consulta** e clique em **OK**.
A formatação dos resultados de consulta é convertida num formulário e aberta numa nova janela de formulário.

Agrupar e agregar resultados de consulta

Pode-se aplicar formatação de agrupamento, agregação e resumo a resultados de consulta.

Agrupar resultados de consulta

Pode-se agrupar resultados de consulta com ou sem informações de resumo.

1. Seleccione a coluna que pretende agrupar.
2. No menu **Resultados**, seleccione o tipo de agrupamento que pretende aplicar.
A coluna é agrupada segundo a sua selecção.

Resumir resultados de consulta

Pode-se resumir resultados de consulta por coluna.

1. Seleccione a coluna que pretende agrupar.
2. No menu **Resultados**, seleccione o tipo de resumo que pretende aplicar.

A coluna é resumida segundo a sua selecção.

Guardar resultados de consulta e formatação

Pode-se guardar resultados de consulta e guardar a formatação como um formulário.

Guardar resultados de consulta como uma tabela

Pode-se guardar resultados de consulta como uma tabela num servidor de base de dados.

1. No menu **Resultados** seleccione **Guardar em Base de Dados**.
Abre-se a caixa de diálogo Guardar Dados.
2. Introduza um proprietário e nome de tabela e clique em **OK**.
Os resultados de consulta serão guardados como uma tabela na base de dados.

Guardar resultados de consulta em ficheiros

Pode-se guardar resultados de consulta num ficheiro no PC ou num servidor de ficheiros.

1. No menu **Resultados** seleccione **Guardar em Ficheiro**.
Abre-se a caixa de diálogo Exportar Dados.
2. Especifique a localização na qual pretende guardar o ficheiro, opções de exportação e clique em **OK**.
Os resultados da consulta são guardados num ficheiro.

Imprimir resultados de consulta

Pode-se pré-visualizar e imprimir os resultados de consulta.

Pré-visualizar resultados de consulta

1. Abra e execute uma consulta. São apresentados os resultados da consulta.
2. No menu **Ficheiro** seleccione **Configurar Página**. Abre-se a caixa de diálogo Configurar Página.
3. Proceda às alterações pretendidas ao esquema da página e clique em **OK**.
4. Clique no botão **Ver Antes** na barra de ferramentas.



Surge uma pré-visualização dos resultados de consulta impressos.

Imprimir resultados de consulta

1. Abra uma consulta e active a vista Resultados. São apresentados os resultados da consulta.

2. No menu **Ficheiro** seleccione **Configurar Página**. Abre-se a caixa de diálogo Configurar Página.
3. Proceda às alterações pretendidas ao esquema da página e clique em **OK**.
4. Clique no botão **Imprimir** na barra de ferramentas.



Os resultados de consulta são impressos.

Capítulo 5. Trabalhar com Relatórios

Criam-se relatórios combinando resultados de consulta com a formatação de um formulário.

Formulários

Os formulários consistem em conjuntos de instruções de formatação utilizadas para criar, apresentar e imprimir relatórios.

Compreender formulários

Os formulários são formados por vários componentes. Todos estes componentes podem ser editados num documento de formulário.

Principal

Os principais componentes de um formulário, incluindo títulos, pés-de-página e quebras.

Quebras

Características, conteúdo e colocação até um máximo de seis linhas de subtotal num relatório.

Cálculos

Expressões de cálculo de relatórios.

Nota: Tem de dispor do ObjectREXX da IBM instalado no seu computador para utilizar cálculos de formulário.

Colunas

A aparência e a formatação de colunas no relatório. As características definíveis incluem ordem, formato, utilização, indentação e largura da coluna.

Condições

Restrições de formatação condicional. Por exemplo, pode-se definir o formulário para não apresentar linhas que não cumpram certas características.

Detalhes

Títulos de detalhe e corpo de texto do relatório. Este é o local onde pode combinar ou substituir dados tabulares por texto de forma livre para criar circulares ou etiquetas de endereço.

Final Conteúdo e colocação do texto final do relatório. Por exemplo, pode optar por incluir texto final e dados de resumo no final do relatório.

HTML

Conteúdo e colocação de identificadores e formatação HTML em relatórios HTML.

Opções

Opções de aparência diversas para o relatório.

Página

Conteúdo e colocação do título e pé-de-página no relatório.

Elaborar um relatório usando um formulário

Criam-se relatórios combinando resultados de consulta com as opções de formatação contidas num formulário. Pode-se elaborar vários relatórios a partir de um único conjunto de resultados repetindo este processo.

1. Numa vista de resultados de consulta, clique no botão **Apresentar Relatório**.



Abre-se a caixa de diálogo Seleccionar Formulário.



2. Ser-lhe-á pedido para fornecer informações adicionais consoante o tipo de formulário que seleccionar na caixa de diálogo Seleccionar Formulário. Especifique a localização do ficheiro, o proprietário e nome ou o título de documento, conforme apropriado, e clique em **OK**. O relatório será criado usando o formulário seleccionado e os actuais resultados de consulta.

Editar um formulário

A janela Formulário faculta muitas opções para edição e formatação de formulários.

Num formulário aberto, visualize o menu Formulário. O menu Formulário enumera todas as opções de edição e formatação do formulário. Também se pode editar qualquer destes componentes fazendo clique no respectivo botão na barra de ferramentas.

Criar um formulário

Todas estas etapas incluem dados exemplo da tabela Q.STAFF. Tente experimentar definições várias para criar os seus próprios formulários personalizados.

Etapa 1: Criar um formulário

1. Execute a seguinte consulta SQL para obter os dados a apresentar no relatório:

```
SELECT * FROM Q.STAFF ORDER BY DEPT, NAME
```

São apresentados os resultados da consulta.

2. Clique no botão **Apresentar Relatório** na barra de ferramentas. Abre-se a caixa de diálogo Seleccionar Formulário.
3. Especifique que pretende usar o formulário predefinido e clique em **OK**. O QMF para Windows apresenta o relatório predefinido. Para alterar o formato predefinido, clique num dos botões de componente de formulário na barra de ferramentas. Será apresentado um botão para cada componente de formulário na barra de ferramentas do formulário.

Etapa 2: Alterar a ordem das colunas

Queremos NAME como primeira coluna no relatório e ID como segunda. A ordem das colunas é especificada no componente Colunas do formulário.

1. Clique em **Colunas...** no menu **Formulário** para apresentar o separador Colunas da caixa de diálogo Formulário.
2. Altere a sequência de uma coluna escrevendo por cima do valor de sequência existente. Para tornar NAME na primeira coluna no relatório, altere o seu número de sequência (a coluna na lista denominado Seq) para 1.
3. Para tornar ID na segunda coluna do relatório, altere o seu número de sequência para 2 e clique em **OK**. O QMF para Windows apresenta o relatório com a nova ordem de colunas na janela Formulário.

Etapa 3: Alterar os títulos de coluna

Queremos que o primeiro título de coluna seja EMPLOYEE e o segundo COMMISSION. O texto dos títulos de coluna é especificado no componente Colunas do formulário.

1. Clique em **Colunas...** no menu **Formulário** para apresentar o separador Colunas da caixa de diálogo Formulário.
2. Altere o título da coluna escrevendo por cima do texto de título existente. Altere o primeiro título de coluna para EMPLOYEE e o último para COMMISSION e clique em **OK**. O QMF para Windows apresenta o relatório com os novos títulos de coluna na janela Formulário.

Etapa 4: Alterar o formato da coluna

Queremos que a coluna SALARY seja apresentada com o símbolo monetário apropriado. O formato da coluna é determinado pelo seu código de edição, o qual está especificado no componente Colunas do formulário.

1. Clique em **Colunas...** no menu **Formulário** para visualizar o separador Colunas da caixa de diálogo Formulário.
2. Altere o código de edição da coluna SALARY para D2 escrevendo por cima do código de edição existente e clique em **OK**. O QMF para Windows apresenta o relatório com a coluna SALARY usando o símbolo monetário apropriado na janela Formulário.

Etapa 5: Adicionar informações de resumo

Queremos dividir o relatório em secções separadas para cada departamento. Além disso, queremos ver os totais de SALARY e COMMISSION relativos a cada departamento no final de cada secção. Neste sentido, precisamos de especificar a utilização pretendida para cada coluna do relatório. A utilização de uma coluna é determinada pelo seu código de utilização e está especificada no componente Colunas do formulário.

1. Clique em **Colunas...** no menu **Formulário** para visualizar o separador Colunas da caixa de diálogo Formulário.
2. Para dividir o relatório em secções baseadas em DEPT, altere o código de utilização de DEPT para BREAK1. Os códigos de utilização começados pela palavra BREAK produzem uma quebra de secção na coluna especificada. O número que se segue à palavra BREAK determina o nível de quebra; é suportado um máximo de seis níveis de quebra num relatório.
3. Para especificar que queremos incluir totais de SALARY e COMMISSION para cada DEPT, altere o código de utilização de SALARY e COMMISSION para SUM.
4. A compreensão do relatório será mais fácil se incluirmos informações descritivas no final de cada quebra de secção. Neste sentido, clique em **Quebras...** no menu **Formulário**.
5. Pode-se especificar texto de pé-de-página de quebra no separador Quebras da caixa de diálogo Formulário. Defina a primeira linha de pé-de-página de quebra para Total de Departamento e clique em **OK**. O QMF para Windows apresenta a janela Formulário.

Etapa 6: Adicionar cabeçalhos e pés-de-página

Queremos adicionar um título e um pé-de-página ao nosso relatório. Os títulos e pés-de-página são especificados no componente Página do formulário.

1. Clique em **Página...** no menu **Formulário** para apresentar o separador Página da caixa de diálogo Formulário.

2. Utiliza-se a parte de cima desta caixa de diálogo para especificar o título de página. Defina a primeira linha do título de página como Relatório de Departamento e a segunda linha como Total de Salário e Comissões. Escolha o alinhamento do título.
3. Utiliza-se a parte de baixo desta caixa de diálogo para especificar o pé-de-página. Defina a primeira linha do pé-de-página como Final de Página. Escolha o alinhamento do pé-de-página e clique em **OK**. O QMF para Windows apresenta a janela Formulário.

Guardar formulários

Pode-se guardar formulários no PC, num servidor de ficheiros ou de base de dados.

Guardar um formulário em ficheiro

1. Num formulário aberto, clique no botão **Guardar**.
2. Se o formulário tiver sido guardado anteriormente, seleccione **Guardar**. Caso contrário, abre-se a caixa de diálogo Guardar Como.
3. Introduza o nome do ficheiro em que pretende armazenar o formulário e clique em **OK**. O formulário será guardado.

Abrir ficheiros de formulários guardados

1. Clique no botão **Abrir** na barra de ferramentas.



Abre-se a caixa de diálogo Abrir.

2. Seleccione o ficheiro que pretende abrir e clique em **OK**. O formulário seleccionado abre-se num novo documento de formulário.

Guardar formulários no servidor de base de dados

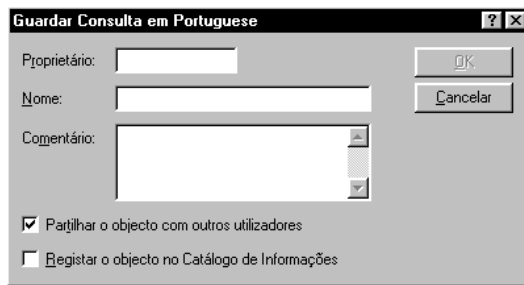
Pode-se disponibilizar o acesso a formulários guardados no servidor a outros utilizadores. Se preferir partilhar formulários com outros utilizadores, guarde-os no servidor de base de dados.

- 1.

Num formulário aberto, clique no botão **Guardar no Servidor** na barra de ferramentas.



Abre-se a caixa de diálogo Guardar Formulário.



2. Introduza um proprietário, um nome, seleccione se pretende ou não partilhar o formulário guardado com outros utilizadores e clique em **OK**. O formulário será guardado no servidor.
Se já existir um formulário com este nome, ser-lhe-á pedido para sobrepor o formulário existente.

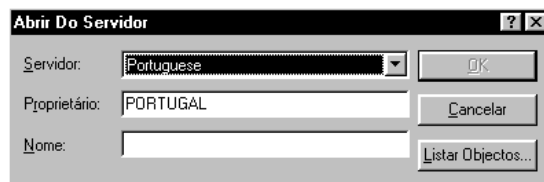
Abrir formulários guardados no servidor de base de dados

Pode-se abrir formulários que tenham sido guardados no servidor de base de dados.

1. Clique no botão **Abrir Do Servidor** na barra de ferramentas.



Abre-se a caixa de diálogo Abrir Do Servidor.



2. Introduza um servidor, proprietário e nome e clique em **OK**. Abre-se o formulário.

Imprimir relatórios

Pode-se elaborar relatórios impressos.

1. Abra um formulário e clique em **Configurar Página**.
2. Proceda às alterações pretendidas ao esquema da página e clique em **OK**.
3. Clique em **Imprimir Relatório** no menu **Ficheiro**.
O formulário será impresso.

Exportar relatórios

Pode-se exportar um relatório para um ficheiro.

1. Abra um formulário e clique em **Configurar Página**.
2. Proceda às alterações pretendidas ao esquema da página e clique em **OK**.
3. Clique em **Exportar Relatório** no menu **Ficheiro**. Abre-se a caixa de diálogo Exportar Relatório.



4. Introduza o nome do ficheiro em que pretende armazenar o relatório e clique em **OK**. O relatório será exportado.

Capítulo 6. Trabalhar com Procedimentos

Os procedimentos lineares permitem-lhe executar consultas, gerar relatórios, editar dados e executar outras funções usando um único comando. Para obter uma listagem completa de todos os comandos de procedimento suportados pelo QMF para Windows, consulte a função de ajuda online.

Os procedimentos com lógica, ou procedimentos REXX, são semelhantes aos procedimentos lineares, mas contêm a linguagem de programação Object REXX da IBM, assim como comandos de procedimento. O Object REXX tem de estar instalado localmente por forma a executar procedimentos com lógica.

Executar procedimentos

Utiliza-se procedimentos para executar funções múltiplas com um único comando.

Criar um novo procedimento linear

A partir do menu **Ficheiro**, seleccione **Novo Procedimento**.

Abre-se um novo documento de procedimento.

Criar um novo procedimento com lógica

1. A partir do menu **Ficheiro**, seleccione **Novo Procedimento**.
Abre-se um novo documento de procedimento.
2. Escreva uma nova linha de comentário REXX como primeira linha do procedimento. As linhas de comentário REXX começam por /* e terminam por */
3. Escreva os comandos de procedimento do QMF pretendidos no procedimento. Os comandos do QMF devem ser inseridos em maiúsculas e delimitados por aspas.
4. Escreva os comandos REXX pretendidos no procedimento.

Nota: Os comandos REXX são executados localmente e não no servidor de base de dados. O Object REXX tem de estar instalado localmente.

Executar um procedimento num servidor de base de dados

1. Abra um novo documento de procedimento e escreva um conjunto de comandos, ou abra um procedimento existente num ficheiro ou no servidor de base de dados.

2. Clique no botão **Executar Procedimento** na barra de ferramentas.



O procedimento é executado.

Guardar procedimentos

Pode-se guardar procedimentos no PC, num servidor de ficheiros ou de base de dados.

Guardar procedimentos em ficheiros

1. Num procedimento aberto, clique no botão **Guardar** da barra de ferramentas.



Mesmo que o procedimento já tenha sido guardado, será guardado novamente. Caso contrário, abre-se a caixa de diálogo Guardar Como.

2. Introduza o nome do ficheiro em que pretende armazenar o procedimento e clique em **OK**. O procedimento será guardado.

Abrir um ficheiro de procedimento guardado

1. Clique no botão **Abrir** na barra de ferramentas.



Surge a caixa de diálogo Abrir.

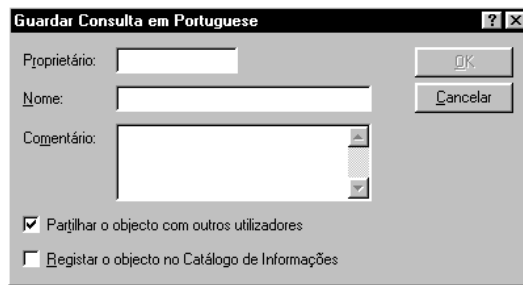
2. Selecciono o ficheiro que pretende abrir e clique em **OK**. O procedimento seleccionado abre-se num novo documento de procedimento.

Guardar um procedimento no servidor de base de dados

1. Num procedimento aberto, clique no botão **Guardar no Servidor** na barra de ferramentas.



Abre-se a caixa de diálogo Guardar Procedimento.



2. Introduza um proprietário, um nome, seleccione se pretende ou não partilhar o procedimento guardado com outros utilizadores e clique em **OK**. O procedimento será guardado no servidor.

Se já existir um procedimento com este nome, ser-lhe-á pedido para sobrepor o procedimento existente.

Abrir procedimentos guardados no servidor de base de dados

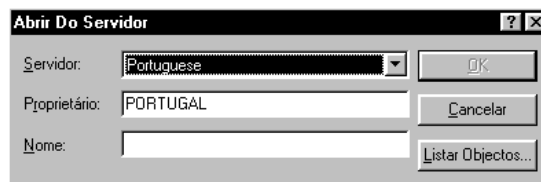
Pode-se abrir procedimentos que tenham sido guardados no servidor de base de dados.

- 1.

Clique no botão **Abrir Do Servidor** na barra de ferramentas.



Abre-se a caixa de diálogo Abrir Do Servidor.



2. Introduza um servidor, proprietário e nome e clique em **OK**. Abre-se o procedimento.

Imprimir procedimentos

Pode-se imprimir o texto de um procedimento.

Pré-visualizar um procedimento

1. Abra um procedimento. Surgem os comandos de procedimento.
2. No menu **Ficheiro** clique em **Configurar Página**. Abre-se a caixa de diálogo Configurar Página.

3. Proceda às alterações pretendidas ao esquema da página e clique em **OK**.
4. Clique no botão **Ver Antes** na barra de ferramentas.



Surge uma pré-visualização do procedimento impresso.

Imprimir um procedimento

1. Abra um procedimento. Surgem os comandos de procedimento.
2. No menu **Ficheiro** clique em **Configurar Página**. Abre-se a caixa de diálogo Configurar Página.
3. Proceda às alterações pretendidas ao esquema da página e clique em **OK**.
4. Clique no botão **Imprimir** na barra de ferramentas.



O procedimento será impresso.

Capítulo 7. Trabalhar com Listas

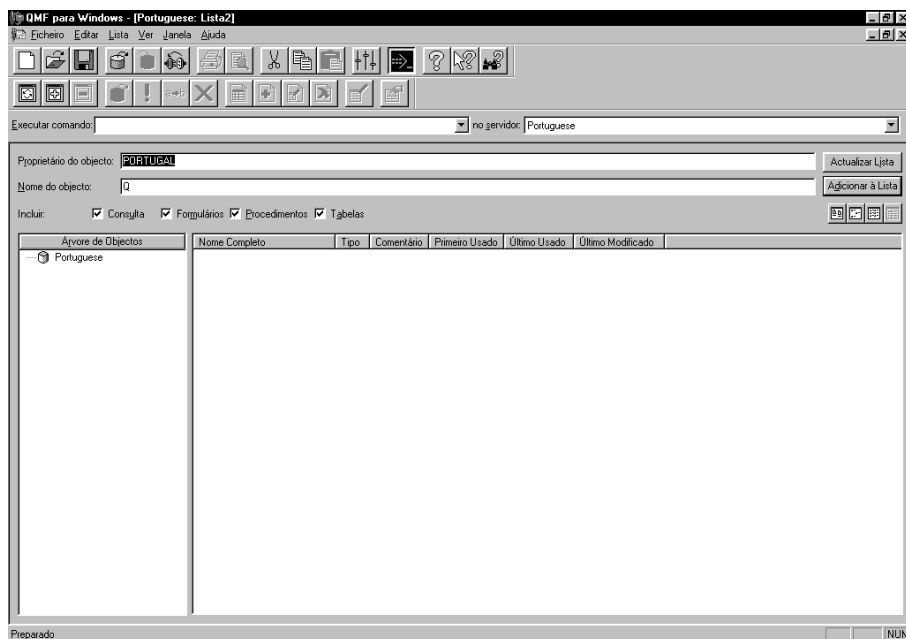
As listas constituem uma forma fácil de ver colecções de objectos do QMF.

Objectos

O QMF para Windows reconhece quatro tipos de objectos: consultas, formulários, procedimentos e tabelas. Pode-se usar a janela Lista para ver os objectos segundo o nome, proprietário e tipo do objecto.

Listar objectos

1. A partir do menu **Ficheiro**, seleccione **Nova Lista**.
Abre-se a janela Lista.



2. Especifique um nome e um proprietário.

Nota: Pode-se usar padrões para seleccionar objectos de uma lista de objectos correspondentes.

- Use o carácter de percentagem (%) para fazer corresponder uma cadeia de qualquer comprimento contendo quaisquer caracteres. Por exemplo, para listar todas as tabelas com um nome começado pela letra A, introduza A%.

- Use o carácter de sublinhado (_) para fazer corresponder um único carácter. Por exemplo, para listar todas as tabelas com um proprietário que tenha a letra A na segunda posição, introduza _A%.
3. Selecciono o tipo de objecto que está a procurar.
 4. Clique em **Actualizar Lista**. Será apresentada uma lista de objectos correspondentes guardados no servidos de base de dados.

Comandos da janela Lista

Ao fazer clique com o botão direito do rato na janela Lista poderá ver uma lista de comandos idênticos aos do menu Lista.

Apresentar objectos

Abre o objecto seleccionado para visualização. Disponível para consultas, formulários, procedimentos e tabelas.

Executar objecto

Executa o objecto seleccionado. Disponível para consultas e procedimentos.

Extrair objecto

Cria uma consulta baseada na tabela seleccionada. Pode optar por extrair uma consulta SQL SELECT, SQL UPDATE, SQL INSERT ou uma consulta orientada. Disponível para tabelas.

Editar objecto

Abre o objecto seleccionado para edição. Disponível para tabelas.

Propriedades

Apresenta as propriedades do objecto seleccionado, incluindo comentários, atributos e informações sobre o histórico de utilização. Disponível para consultas, formulários, procedimentos e tabelas.

Criar listas

Pode-se criar listas para servirem de colecções de objectos. Por exemplo, poder-se-ia criar uma lista de todas as consultas, formulários, procedimentos e tabelas relacionadas com inventário para manter o trabalho num só local. Uma vez criadas, pode-se adicionar e remover objectos das listas e guardá-las para utilização futura.

Adicionar objectos a listas

Pode-se adicionar objectos a listas.

Numa lista aberta, especifique as informações de nome e proprietário dos objectos que pretende adicionar e clique no botão **Adicionar à Lista** na barra

de ferramentas.



Os objectos que correspondam ao nome e ao proprietário serão adicionados á lista.

Remover objectos de listas

Pode-se remover objectos não relacionados de listas.

Numa lista aberta, clique no botão **Remover** da barra de ferramentas.



O objecto é removido da lista mas não é eliminado.

Guardar listas em ficheiros

1. Numa lista aberta, clique no botão **Guardar** da barra de ferramentas.



Mesmo que a lista já tenha sido guardada, será novamente guardada. Caso contrário, abre-se a caixa de diálogo Guardar Como.

2. Introduza o nome do ficheiro em que pretende armazenar a lista e clique em **OK**. A lista será guardada.

Abrir ficheiros de lista guardados

1. Clique no botão **Abrir** na barra de ferramentas.



Abre-se a caixa de diálogo Abrir.

2. Selecciono o ficheiro que pretende abrir e clique em **OK**. A lista seleccionada abre-se num documento de lista.

Capítulo 8. Trabalhar com Ficheiros de Trabalho

Pode-se marcar e executar procedimentos usando ficheiros de trabalho. Os ficheiros de trabalho utilizam o programador do Windows para executar procedimentos segundo a data e a hora predefinidas pelo utilizador.

Ficheiros de trabalho

Pode-se criar ficheiros de trabalho e armazená-los localmente ou no servidor de base de dados.

Criar ficheiros de trabalho

1. A partir do menu **Ficheiro**, seleccione **Novo Trabalho**.
Abre-se um novo documento de trabalho.

Executar ficheiros de trabalho

Pode-se executar ficheiros de trabalho que tenham sido guardadas localmente.

1. Abra um ficheiro de trabalho.
2. Clique no botão **Executar Trabalho** na barra de ferramentas.



3. Arraste a linha divisória de um lado a outro ou para cima e para baixo para redimensionar a coluna ou a linha.

Nota: Se guardar a consulta depois de redimensionar linhas e colunas, a consulta será sempre apresentada com a nova formatação.

Ajustar automaticamente colunas e linhas

Pode-se dimensionar automaticamente colunas e linhas para se ajustarem aos dados que contém.

Selecione, com o rato, uma coluna ou linha inteira e clique duas vezes na linha divisória negra entre ela e o objecto adjacente. A coluna ou a linha será automaticamente redimensionada para se ajustar aos dados.

Nota: Se guardar a consulta depois de redimensionar linhas e colunas, a consulta será sempre apresentada com a nova formatação.

Ordenar resultados de consulta

Uma vez a consulta executada, pode-se dispor os resultados em ordem alfabética por coluna.

Na vista Resultados de uma consulta, selecione uma coluna e selecione **Ordenação Ascendente** no menu **Resultados**.

Os resultados da consulta são dispostos por ordem ascendente.

-ou-

Na vista Resultados de uma consulta, selecione uma coluna e selecione **Ordenação Descendente** no menu **Resultados**.

Os resultados da consulta são dispostos por ordem descendente.

Nota: Para aplicar uma ordenação mais complexa à coluna seleccionada, selecione **Ordenar** no menu **Resultados**.

Reordenar colunas

Pode-se alterar a ordem das colunas em resultados de consulta.

Na vista Resultados de uma consulta, selecione uma coluna e arraste-a para a nova posição.

A coluna aparece na nova ordenação.

Formatar resultados de consulta

Pode-se alterar o tipo de letra utilizado para apresentar consultas e respectivos resultados. A escolha de tipos de letra varia consoante os elementos instalados no computador. Para mais informações sobre a adição de tipos de letra, consulte a função da ajuda do sistema operativo.

Nota: Se guardar a consulta depois de seleccionar um novo tipo de letra para apresentação dos resultados, estes sê-lo-ão sempre usando o novo tipo de letra.

Seleccionar o tipo de letra de apresentação dos resultados de consulta

1. Na vista Resultados, selecione **Definir Tipo de Letra** no menu **Resultados**. Abre-se a caixa de diálogo Tipos de Letra.
2. Selecione o tipo e o tamanho de letra para apresentação dos resultados da consulta e clique em **OK**. Os resultados da consulta serão apresentados no formato que especificou.

Nota: Clique em **Predefinir** para utilizar o tipo de letra seleccionado como predefinição para todos os resultados de consultas.

Formatar resultados de consulta numéricos

1. Na vista Resultados, seleccione uma coluna contendo valores numéricos e seleccione **Formatar** no menu **Resultados**. Abre-se a caixa de diálogo **Formatar**.
2. Especifique a formatação que pretende aplicar e clique em **OK**. Os valores

Nota: Clique em **Predefinir** para utilizar o tipo de letra seleccionado como predefinição para todos os resultados de consultas.

Converter a formatação de resultados de consulta num formulário

Pode-se converter a formatação de resultados de consulta num formulário.

1. No menu **Resultados** seleccione **Apresentar Relatórios**.
Abre-se a caixa de diálogo **Selecionar Formulário**.
2. Seleccione **Da Consulta** e clique em **OK**.
A formatação dos resultados de consulta é convertida num formulário e aberta numa nova janela de formulário.

Agrupar e agregar resultados de consulta

Pode-se aplicar formatação de agrupamento, agregação e resumo a resultados de consulta.

Agrupar resultados de consulta

Pode-se agrupar resultados de consulta com ou sem informações de resumo.

1. Seleccione a coluna que pretende agrupar.
2. No menu **Resultados**, seleccione o tipo de agrupamento que pretende aplicar.
A coluna é agrupada segundo a sua selecção.

Resumir resultados de consulta

Pode-se resumir resultados de consulta por coluna.

1. Seleccione a coluna que pretende agrupar.
2. No menu **Resultados**, seleccione o tipo de resumo que pretende aplicar.
A coluna é resumida segundo a sua selecção.

Guardar resultados de consulta e formatação

Pode-se guardar resultados de consulta e guardar a formatação como um formulário.

Guardar resultados como uma tabela

Pode-se guardar resultados de consulta como uma tabela num servidor de base de dados.

1. No menu **Resultados** seleccione **Guardar em Base de Dados**.

Abre-se a caixa de diálogo Guardar Dados.

2. Introduza um proprietário e nome de tabela e clique em **OK**.
Os resultados de consulta serão guardados como uma tabela na base de dados.

Guardar resultados de consulta em ficheiros

Pode-se guardar resultados de consulta num ficheiro no PC ou num servidor de ficheiros.

1. No menu **Resultados** seleccione **Guardar em Ficheiro**.
Abre-se a caixa de diálogo Exportar Dados.
2. Especifique a localização na qual pretende guardar o ficheiro, opções de exportação e clique em **OK**.
Os resultados da consulta são guardados num ficheiro.

Imprimir resultados de consulta

Pode-se pré-visualizar e imprimir os resultados de consulta.

Pré-visualizar resultados de consulta

1. Abra e execute uma consulta. São apresentados os resultados da consulta.
2. No menu **Ficheiro** seleccione **Configurar Página**. Abre-se a caixa de diálogo Configurar Página.
3. Proceda às alterações pretendidas ao esquema da página e clique em **OK**.
4. Clique no botão **Ver Antes** na barra de ferramentas.



Surge uma pré-visualização dos resultados de consulta impressos.

Imprimir resultados de consulta

1. Abra uma consulta e active a vista Resultados. São apresentados os resultados da consulta.
2. No menu **Ficheiro** seleccione **Configurar Página**. Surge a caixa de diálogo Configurar Página.
3. Proceda às alterações pretendidas ao esquema da página e clique em **OK**.
4. Clique no botão **Imprimir** na barra de ferramentas.



Os resultados de consulta são impressos.

Capítulo 9. Trabalhar com consultas estáticas

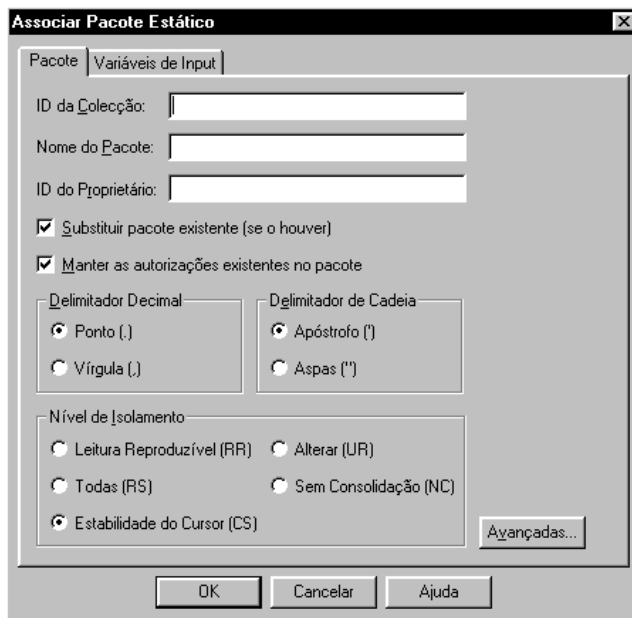
Uma consulta estática consiste numa consulta SQL que tenha sido anteriormente transmitida a um servidor de base de dados e associada num pacote. Ao executar uma consulta estática, o servidor de base de dados utiliza o texto SQL associado no pacote, ao invés do texto SQL que surge actualmente na janela de consulta. As consultas estáticas são mais eficientes, em termos de recursos, do que as consultas dinâmicas, mas as primeiras não podem ser editadas.

Consultas estáticas

As consultas estáticas são criadas a partir de consultas SQL e orientadas já existentes.

Criar consultas estáticas

1. A partir do menu **Consulta**, seleccione **Associar Pacote Estático**. Abre-se a caixa de diálogo Associar Pacote Estático.



2. Seleccione o separador Pacote, introduza um ID de colecção e nome de pacote e altere quaisquer outras opções pretendidas.

3. Se a consulta contiver variáveis de substituição, selecione o separador Variáveis. Substitua quaisquer variáveis de substituição por variáveis de sistema central.
4. Clique em **OK**. A consulta estática será associada.

Nota: Depois de associar uma consulta terá de a guardar também, quer em ficheiro quer no servidor de base de dados.

Substituir variáveis de substituição por variáveis de sistema central

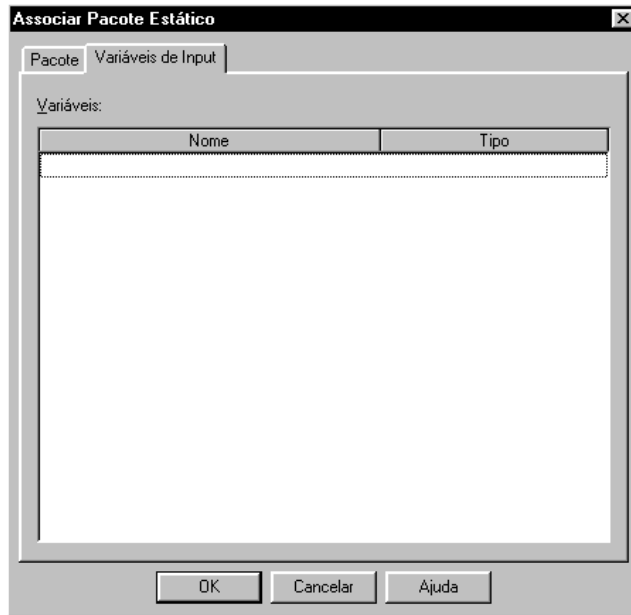
Ao associar um pacote, tem de especificar uma variável de sistema central para utilizar em vez de cada variável de substituição no texto SQL. Todavia, nem sempre se pode substituir uma variável de substituição directamente por uma variável de sistema central. As variáveis de substituição fornecem substituição de texto directa no texto da consulta antes de o texto ser enviado ao servidor de base de dados. As variáveis de sistema central são enviadas ao servidor de base de dados como parte da consulta. Consulte a documentação do servidor da sua base de dados acerca de regras sobre a utilização de variáveis de sistema central em consultas.

Uma vez especificada a relação entre uma variável de substituição e outra de sistema central, o QMF para Windows recorda a relação e sugere-a em futuras consultas, tornando assim mais simples a associação de pacotes.

Os tipos de dados válidos para variáveis de sistema central são os seguintes:

- CHAR(n)
- VARCHAR(n)
- INTEGER
- SMALLINT
- FLOAT
- DECIMAL(p,s)
- DATE
- TIME
- TIMESTAMP

1. Na caixa de diálogo Associar Pacote Estático seleccione o separador Variáveis de Input.



2. Introduza o tipo de variável para cada variável de sistema central e clique em **OK**. As variáveis de substituição são convertidas em variáveis de sistema central.

Executar uma consulta estática

As consultas estáticas executam-se da mesma forma que quaisquer outras consultas. Consulte "Consultas SQL" na página "Consultas SQL" na página 9.

Capítulo 10. Trabalhar com o Editor de Tabelas

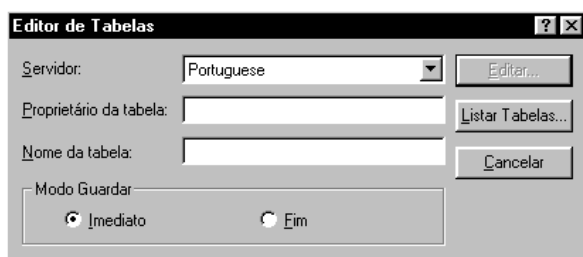
Utilize o editor de tabelas para procurar, adicionar, editar ou eliminar dados armazenados em tabelas sem ter de escrever instruções SQL.

Editor de tabelas

O editor de tabelas permite flexibilidade ao editar e procurar dados.

Procurar linhas usando o editor de tabelas

1. A partir do menu **Ficheiro**, seleccione **Editor de Tabelas**. Abre-se a caixa de diálogo Editor de Tabelas.



2. Especifique uma tabela.

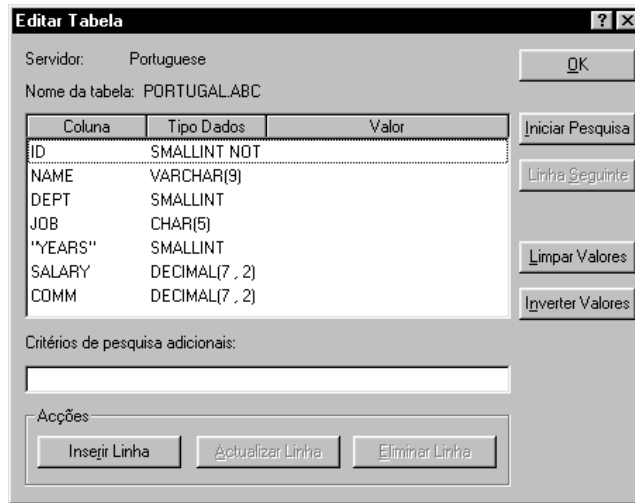
Nota: Pode-se usar padrões para seleccionar nomes de tabela de uma lista de tabelas correspondentes.

- Use o carácter de percentagem (%) para fazer corresponder uma cadeia de qualquer comprimento contendo quaisquer caracteres. Por exemplo, para listar todas as tabelas com um nome começado pela letra A, introduza A%.
- Use o carácter de sublinhado (_) para fazer corresponder um único carácter. Por exemplo, para listar todas as tabelas com um proprietário que tenha a letra A na segunda posição, introduza _A%.

Depois de inserir o padrão, clique em **Listar Tabelas** e seleccione uma tabela da lista resultante.

3. Seleccione um Modo de Guardar.
 - **Imediato** - A tabela é actualizada no servidor de base de dados imediatamente após cada alteração.
 - **Final** - A tabela é actualizada no servidor de base de dados depois de terminar de inserir todas as alterações. Os outros utilizadores não

poderão efectuar alterações enquanto estiver a proceder às suas.



4. Clique em **Editar**. Abre-se a caixa de diálogo Editar tabela.
5. Introduza os valores que pretende procurar na coluna Valor ou escreva critérios de procura no campo Critérios de procura adicionais para especificar condições de procura mais complexas. Pode-se introduzir qualquer predicado SQL válido no campo Critérios de procura adicionais.
6. Clique em **Iniciar Procura**. A primeira linha correspondente é apresentada na coluna Valor.

Adicionar uma linha

1. Na caixa de diálogo Editar Tabela, introduza a informação relativa ao novo registo.
2. Clique em **Inserir Linha**. A nova linha é adicionada à tabela.
3. Clique em **OK**. As alterações serão guardadas.

Alterar uma linha

1. Na caixa de diálogo Editar Tabela, procure a linha que pretende alterar.
2. Clique em **Linha Seguinte** até ser apresentada a linha que pretende alterar.
3. Edite os dados na coluna Valor e clique em **Actualizar Linha**. A linha é actualizada.
4. Clique em **OK**. As alterações serão guardadas.

Eliminar uma linha

1. Na caixa de diálogo Editar Tabela, procure a linha que pretende eliminar.
2. Clique em **Linha Seguinte** até ser apresentada a linha que pretende eliminar.

3. Clique em **Eliminar Linha**. A linha é eliminada.
4. Clique em **OK**. As alterações serão guardadas.

Editar tabelas na vista de resultados de consulta

Pode-se editar tabelas directamente na vista de resultados de consulta.

Eliminar uma linha da vista de resultados de consulta

Pode-se eliminar linhas individuais em tabelas na vista de resultados de consulta.

Na vista de resultados de consulta, seleccione uma linha e seleccione **Eliminar** no menu **Editar**. A linha é eliminada.

Actualizar colunas na vista de resultados de consulta

Pode-se actualizar o conteúdo de colunas individuais na vista resultados de consulta.

Na vista de resultados de consulta, faça duplo clique numa célula, introduza um novo valor e prima Enter. A tabela é actualizada.

Formulários de DB2

Se dispuser do componente DB2 Forms User instalado no computador, poderá utilizá-lo como editor de tabelas relativamente a tabelas que não contenham dados LOB. Para obter mais informações sobre Formulários do DB2, consulte o Resource Center for DB2 Forms em www.rocketsoftware.com/db2forms.

Capítulo 11. Distribuir Dados

Pode-se exportar dados para outras bases de dados e aplicações.

Exportar dados

Pode-se exportar dados do QMF para Windows para outras aplicações das seguintes formas:

- Exporte os dados para um ficheiro de Texto, CSV, IXF, ou HTML
- Guarde resultados de consulta numa tabela
- Adicione resultados de consulta directamente a uma folha de cálculo do Microsoft Excel

Exportar dados para ficheiros

1. Ao visualizar resultados de consulta, seleccione **Exportar Dados** no menu **Ficheiro**. Abre-se a caixa de diálogo Exportar Dados.



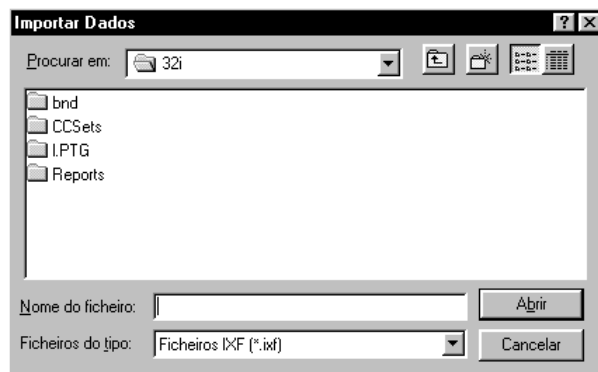
2. Seleccione o Tipo de Ficheiro de Output pretendido e clique no botão **Opções**. Consoante o tipo de ficheiro de output seleccionado, surgirá a caixa de diálogo Exportar Texto/Opções DEL, a caixa de diálogo Exportar Opções HTML, a caixa de diálogo Exportar Opções IXF ou a caixa de diálogo Exportar Opções CSV.

- Pode-se elaborar um ficheiro de texto com uma extensão .TXT. Trata-se de um ficheiro ASCII standard com delimitadores de coluna e cadeia opcionais (tal como especificado na caixa de diálogo Exportar Texto/Opções DEL).
 - Pode-se elaborar um ficheiro HTML com uma extensão .HTM. Trata-se de um ficheiro HTML que pode ser visualizado por qualquer browser da web. Todos os identificadores HTML serão automaticamente gerados no ficheiro, ficando assim pronto a ser publicado no seu site da web da Internet ou de uma intranet. As opções que escolher na caixa de diálogo Exportar Opções HTML controlam a aparência dos dados exportados.
 - Pode-se elaborar um ficheiro .IXF. Uma exportação IXF permite preservar todas as informações da base de dados, incluindo títulos de coluna e tipos de dados. Utiliza-se normalmente para transferir informações de uma base de dados para outra.
 - Pode-se elaborar um ficheiro .CSV. Uma exportação CSV é muito semelhante a uma exportação de texto, utilizando uma vírgula como delimitador de coluna. Este formato usa-se normalmente em aplicações de folha de cálculo.
3. Selecciona as opções para o tipo de ficheiro de exportação seleccionado e clique em **OK**. Fecha-se a caixa de diálogo Opções.
 4. Clique em **OK** na caixa de diálogo Exportar Dados. Os dados serão exportados.

Importar dados

Pode-se importar dados que tenham sido guardados num ficheiro IXF. Uma vez os dados importados para uma janela de consulta, podem ser guardados num servidor de base de dados, exportados para um novo ficheiro ou utilizados para relatórios. São suportados ficheiros PCIXF e System370 IXF de modo de caracteres.

1. A partir do menu **Ficheiro**, seleccione **Importar Dados**. Abre-se a caixa de diálogo Importar Dados.

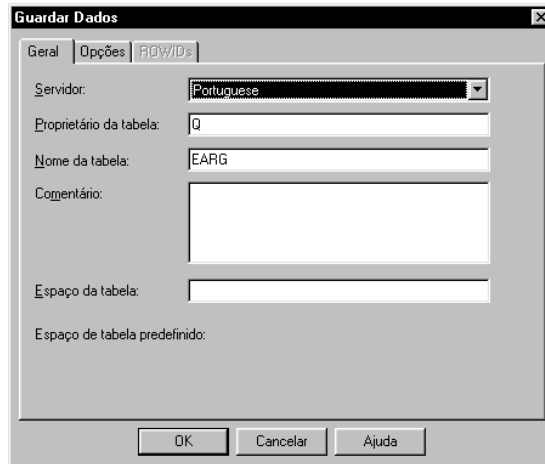


2. Seleccione o ficheiro que pretende importar e clique em **OK**. Os dados importados serão apresentados numa nova janela de consulta.

Guardar dados num servidor de base de dados

Pode-se guardar resultados de consulta importados para uma tabela de base de dados.

1. Ao visualizar resultados de consulta importados, seleccione **Guardar Dados** no menu **Ficheiro**. Abre-se a caixa de diálogo Guardar Dados.



2. Seleccione um servidor de base de dados, introduza um proprietário e nome de tabela, escolha quaisquer outras opções pretendidas e clique em **OK**. Os dados serão guardados.

Utilizar o comando Enviar Para

O QMF para Windows inclui um comando Enviar Para e um cliente básico de correio electrónico. Pode-se utilizar o comando Enviar Para To em conjunto com ficheiros de trabalho para marcar consultas e distribuir os resultados.

1. A partir do menu **Ficheiro**, seleccione **Enviar Para** e **Correio da Internet**. Abre-se a caixa de diálogo Mensagem.
2. Especifique um destinatário da mensagem, um assunto, o texto da mensagem e clique em **Seguinte**. Abre-se a caixa de diálogo Anexos.
3. Adicione ou remova anexos à mensagem e clique em **Seguinte**. Abre-se a caixa de diálogo Enviar Mensagem.
4. Especifique o nome do servidor de correio e clique em **Terminar**. A mensagem é enviada.

Utilizar o Suplemento para Microsoft Excel

O QMF para Windows inclui um suplemento para Microsoft Excel 7.0 ou posterior. Estes suplementos permitem-lhe executar o QMF para Windows a partir do Excel e devolver resultados de consulta directamente numa folha de cálculo. O suplemento apropriado é automaticamente instalado se escolher a opção de instalação "Normal" ou se escolher a opção de instalação "Personalizada" e seleccionar a opção Suplemento para Microsoft Excel.

1. Clique no botão **QMF para Windows** na barra de ferramentas do Excel.



Abre-se o QMF para Windows.

2. No QMF para Windows, seleccione e execute uma consulta. São apresentados os resultados da consulta.
3. Seleccione os dados que pretende devolver em Excel.
4. A partir do menu **Ficheiro**, seleccione **Devolver dados para Microsoft Excel**. O Excel abre e apresenta a caixa de diálogo Suplemento do QMF para Windows.
5. Introduza o intervalo de destino para os dados e clique em **OK**. Os dados serão adicionados à folha de cálculo.

Utilizar Aplicações Exemplo

Estão disponíveis várias aplicações exemplo e soluções de integração para o QMF para Windows. Visite o site da IBM na Web em <http://www.ibm.com/qmf/> para obter mais informações.

Capítulo 12. Utilizar o Centro de Relatórios do QMF

O Centro de Relatórios do QMF permite-lhe elaborar relatórios personalizados usando consultas, formulários, procedimentos e tabelas partilhados do QMF. Com o acesso rápido a estes objectos, pode-se especificar preferências de formato de dados e elaborar relatórios personalizados que podem ser vistos e manuseados em diversas aplicações.

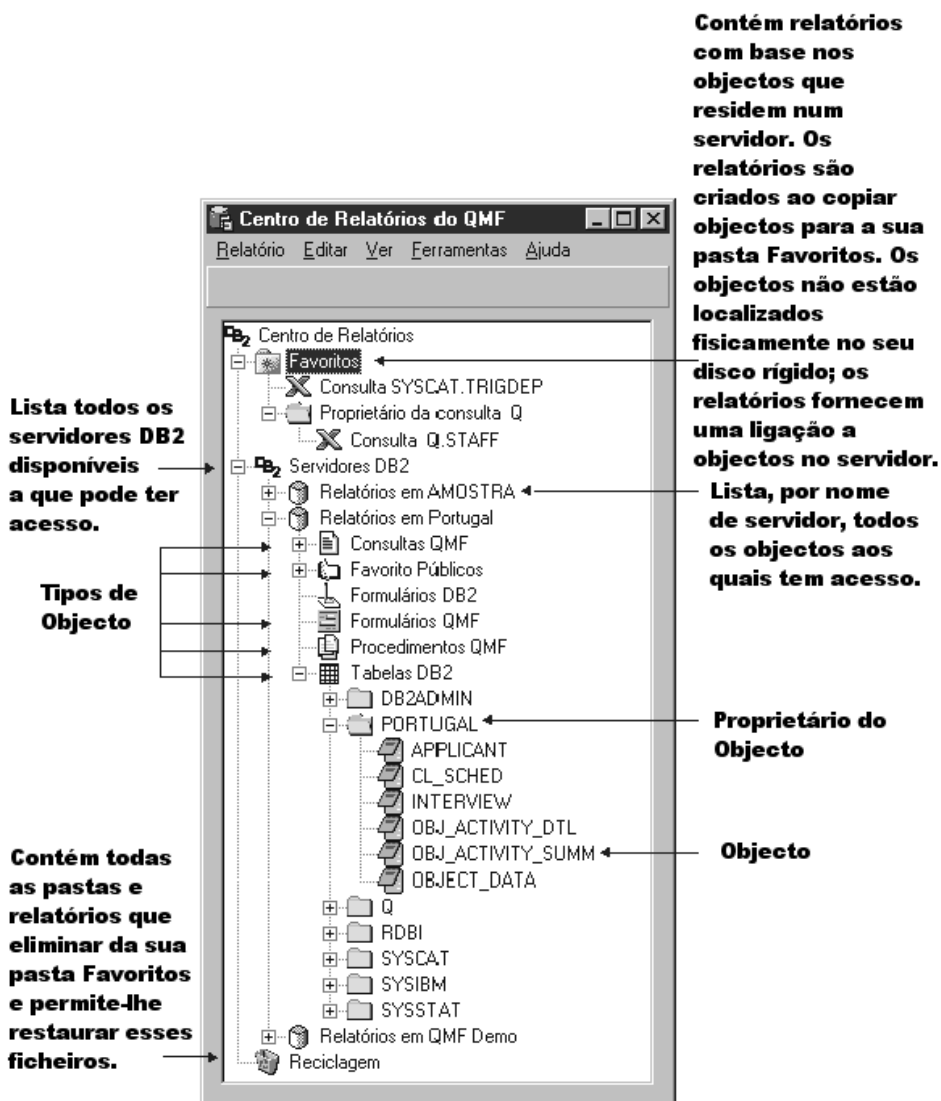
Iniciação ao Centro de Relatórios do QMF

- Clique com o botão direito do rato em qualquer objecto ou pasta para activar as mesmas opções disponíveis nos menus da barra de ferramentas.
- Clique no sinal de mais (+) junto a qualquer pasta para abrir o primeiro nível de conteúdo. Mantenha a tecla SHIFT premida enquanto clica no sinal de mais (+) para abrir todos os níveis abaixo dessa pasta.

Janela do Centro de Relatórios do QMF

A janela do Centro de Relatórios do QMF contém uma estrutura tipo árvore de Favoritos disponíveis, servidores DB2, Favoritos Públicos, objectos e uma

Reciclagem.

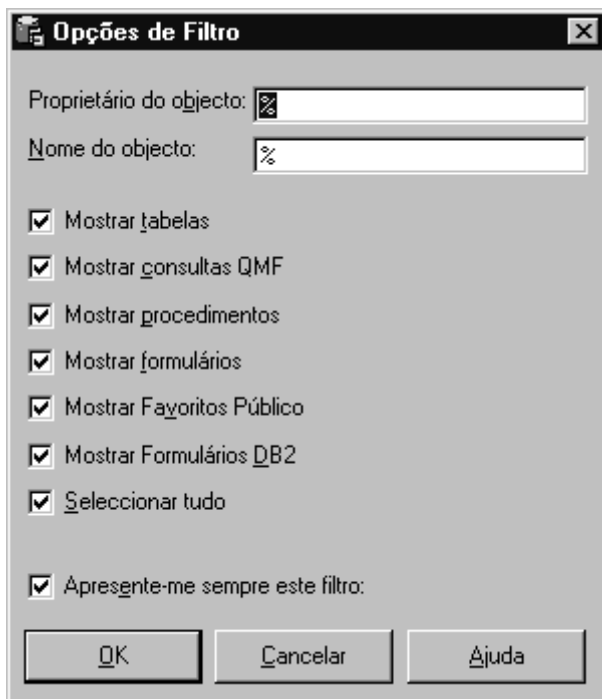


Os objectos apresentados nesta janela incluem um ícone que representa o tipo de aplicação ao qual está associado o output do objecto.

Ligar ao Servidor

1. Se não estiverem apresentados nomes de servidor abaixo de Servidores DB2, clique no sinal de mais (+).

2. Clique no sinal de mais (+) junto a um servidor. Abre-se a caixa de diálogo Opções de Filtragem.



3. Selecciona os tipos de objecto que pretende ver e clique em **OK**. São apresentados os objectos disponíveis no servidor, agrupados por tipo de objecto.

Trabalhar com Relatórios e Objectos

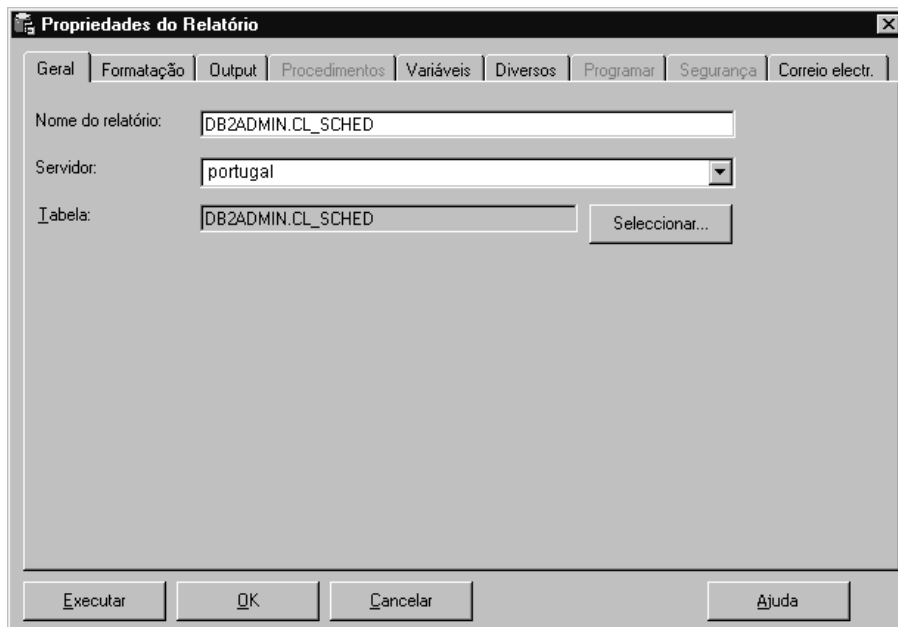
Os relatórios têm por base objectos do QMF. Todos os itens contidos nas suas pastas Favoritos Públicos e Pessoais são considerados relatórios; pode-se manipular as opções de formatação e apresentação inerentes a estes itens. Os itens contidos nestas pastas Favoritos remetem para os objectos do QMF que residem no servidor. Ao efectuar alterações, estas incidem não no objecto do QMF, mas sim na ligação ao objecto que é referido como relatório. Dado que os relatórios têm por base objectos, as propriedades destes últimos também se aplicam aos relatórios.

Pode-se criar relatórios a partir de objectos que residam num servidor; porém, estes não são guardados no servidor. Esta função permite-lhe criar rapidamente relatórios únicos. Contudo, depois de criar um relatório a partir de objectos num servidor, é-lhe dada a opção de guardar o relatório na sua pasta Favoritos.

Executar Relatórios

Pode-se executar relatórios a partir das pastas Favoritos ou de objectos localizados no servidor.

1. Com o relatório ou o objecto seleccionado, escolha *Propriedades* no menu Relatório. Abre-se a caixa de diálogo Propriedades de Relatório.



2. Defina as propriedades, se assim o entender.
3. Clique no botão **Executar**. O relatório é processado e, caso tenha seleccionado a opção *Ver relatório depois da publicação* na caixa de diálogo Output de Propriedades de Relatório, este será apresentado na aplicação que tiver especificado.

Também se pode executar rapidamente um relatório com um dos seguintes métodos:

- Seleccione o relatório e depois *Executar* no menu Relatório.
- Clique com o botão direito do rato no relatório e depois seleccione *Executar*.
- Clique duas vezes no nome do relatório.

Trabalhar com Pastas e Favoritos

Utilizam-se pastas para agrupar relatórios e objectos do QMF; as pastas são denominadas segundo os nomes dos proprietários dos objectos. Pode-se efectuar as mesmas operações em pastas como em relatórios, tais como executar relatórios e definir propriedades de relatório. Ao efectuar tais

operações numa pasta, estas incidirão em cada relatório contido na dita pasta. Por exemplo, se preferir executar consecutivamente cada relatório contido numa pasta, seleccione-a e em seguida escolha **Executar** no menu Relatório.

O Centro de Relatórios do QMF contém duas pastas de nível superior nas quais se podem armazenar relatórios. As pastas contêm relatórios que remetem para objectos em servidores; os próprios objectos não estão contidos nas pastas Favoritos. A sua pasta Favoritos Pessoais reside localmente (no seu PC), de modo que é o único utilizador a poder aceder à pasta e ao seu conteúdo. A pasta Favoritos Públicos reside no servidor e pode ser acedida por todos os utilizadores autorizados. Poderá ter acesso a várias pastas Favoritos Públicos, consoante os seus limites de recurso, ainda que nunca exista mais do que uma pasta Favoritos Públicos em cada servidor.

Ao copiar objectos do QMF para as pastas Favoritos, as pastas são automaticamente renomeadas para incluir o tipo de objecto e o nome de proprietário. Ao copiar um tipo de objecto inteiro (ou seja, uma pasta de objectos do mesmo tipo) de um servidor (p.ex., todas as consultas), o nome do servidor também é incluído no novo nome da pasta.

Adicionar Relatórios a Favoritos

Pode adicionar qualquer objecto ou relatório de um servidor à sua pasta Favoritos Pessoais ou à pasta Favoritos Públicos num servidor (desde que lhe tenha sido concedida permissão pelo administrador de sistema).

Para adicionar relatórios a Favoritos pessoais:

Com o relatório ou o objecto seleccionado, escolha *Adicionar aos favoritos* no menu Relatório, ou arraste o relatório ou o objecto à sua pasta Favoritos pessoais. É adicionado um relatório ao topo da sua pasta Favoritos pessoais com a seguinte convenção de nomenclatura:
TipoobjectoNOMEPROPRIETÁRIO.NOMEOBJECTO.

Para adicionar relatórios a Favoritos Públicos:

Arraste o objecto do QMF ou o relatório para a pasta Favoritos Públicos num servidor. Pode adicionar relatórios a partir da sua pasta Favoritos pessoais ou de qualquer servidor.

Nota: Ao adicionar aos Favoritos Públicos ou ao modificar relatórios nestes últimos, tem de seleccionar *Guardar alterações aos Favoritos Públicos* no menu Relatório antes de as actualizações serem guardadas no servidor.

Para mais informações sobre a utilização do Centro de Relatórios do QMF, consulte o sistema de ajuda online.

Capítulo 13. Utilizar a API do QMF para Windows

Pode-se criar aplicações personalizadas com a interface de programação de aplicações do QMF para Windows.

Controlar o QMF para Windows através da API

Os passos seguintes fornecem uma descrição geral da utilização da API para controlar o QMF para Windows.

1. Crie uma instância do objecto da API do QMF para Windows. Se utilizar o Microsoft Visual Basic, adicione uma referência à biblioteca de tipos do QMF para Windows, qmfwin.tlb. Em seguida utilize a instrução Dim:

```
Dim QMFWin As New QMFWin
```

Ou a instrução CreateObject:

```
Dim QMFWin As Object
```

```
Set QMFWin = CreateObject ("QMFWin.Interface")
```

Nota: Se utilizar um ambiente de programação diferente, consulte a documentação inerente ao produto para saber como proceder.

2. Selecciono o servidor de DB2 que pretende utilizar e chame InitializeServer() para iniciar ligação à base de dados.

Nota: Só se pode iniciar um servidor depois de o DB2 validar um ID de utilizador e uma palavra-passe. Pode fazer com que o QMF para Windows peça o ID de utilizador e a palavra-passe, ou pode pedi-los na sua aplicação e transmiti-los como parâmetros na função de chamada InitializeServer().

3. Selecciono a consulta que pretende executar utilizando InitializeQuery(). Se a consulta contiver variáveis, utilize a função SetVariable() para definir os valores de variável.
4. Abra ou execute a consulta. Utilize a função Open() para abrir o cursor da consulta a instruções SELECT e utilize a função Execute() para executar as instruções SQL diferentes de SELECT.
5. Se a consulta for uma instrução SELECT, busque linhas de dados chamando repetidamente FetchNextRow(). Para buscar mais do que uma linha de uma vez, utilize FetchNextRows(), ou utilize CompleteQuery() para instruir o QMF para Windows no sentido de buscar todas as linhas.
6. Se a consulta for uma instrução SELECT, feche a consulta com a função Close().

7. Termine a unidade de trabalho com as funções Commit() ou Rollback().

Bloquear chamadas

Todas as funções da API do QMF para Windows são síncronas, o que significa que ficam bloqueadas, ou não devolvem, até ser concluída a acção de base de dados solicitada. Esta implementação é preferível porque simplifica a programação na aplicação cliente. Todavia, se a aplicação cliente for de sessão única, não poderá responder a input de utilizador nem efectuar actualizações de ecrã enquanto aguarda pela devolução de uma função da API do QMF para Windows.

A API do QMF para Windows responde a uma chamada de função de cliente de cada vez. Se a aplicação cliente for de sessões múltiplas, terá de:

- aguardar a conclusão de uma função de chamada antes de efectuar outra, ou
- criar instâncias múltiplas da API do QMF para Windows (uma para cada sessão que utilize a API).

Ligar à base de dados

Cada instância do objecto da API do QMF para Windows cria e usa uma única ligação à base de dados para toda a actividade de base de dados que esteja sujeita a subsequente consolidação ou remoção de alterações, incluindo abrir consulta, buscar dados e executar instruções SQL.

Se criar mais do que uma consulta numa dada instância do objecto da API do QMF para Windows chamando InitializeQuery() duas ou mais vezes, todas as consultas irão partilhar a mesma ligação única.

A API do QMF para Windows responde a uma chamada de função de cliente de cada vez. Se a aplicação cliente for de sessões múltiplas, terá de executar:

- DeleteQMFObject()
- GetQMFObjectInfo()
- GetQMFObjectInfoEx()
- GetQMFObjectList()
- GetQMFObjectListEx()
- GetQMFQueryText()
- SaveQMFQuery()

O QMF para Windows cria e usa uma segunda ligação à base de dados por forma a processar actividade administrativa de base de dados (por exemplo, obter informações sobre o QMF). Esta segunda ligação é necessária para suportar um mecanismo coerente de remoção de alterações e de consolidação relativamente a aplicações cliente.

O objecto da API do QMF para Windows processa automaticamente estas ligações à base de dados. Contudo, se o administrador de sistema tiver estabelecido um limite de ligações permitidas, lembre-se de que cada instância do objecto da API do QMF para Windows pode utilizar duas ligações.

Referência API

Esta referência enumera todos os comandos disponíveis para criação de aplicações usando a API do QMF para Windows.

AddDecimalHostVariable()

AddDecimalHostVariable curto(*QueryID* longo, *Type* curto, *Precision* curto, *Scale* curto, VARIANT&*Value* const)

Descrição

Esta função aplica os dados em *Value* a uma variável na instrução SQL estática iniciada com *QueryID*. Chama-se esta função para cada variável na instrução. O QMF para Windows não tenta combinar valores com variáveis, de forma que tem de chamar esta função pela mesma ordem das variáveis na instrução SQL.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por InitializeStaticQuery().
<i>Type</i>	O tipo de dados SQL do valor a transmitir ao servidor de base de dados. Este valor influencia a conversão de <i>Value</i> de um tipo de dados VARIANT para o valor realmente transmitido. O único valor válido para AddDecimalHostVariable() é 484 (RSDT_DECIMAL).
<i>Precision</i>	A precisão do valor decimal.
<i>Scale</i>	A escala do valor decimal.
<i>Value</i>	O valor de dados a substituir na instrução. Para especificar um valor nulo, o tipo da variante deve ser definido como VT_EMPTY.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType() para obter informações de erro adicionais.

AddHostVariable()

AddHostVariable curto(*QueryID* longo, *Type* curto, VARIANT& *Value* const)

Descrição

Esta função aplica os dados em *Value* a uma variável na instrução SQL estática iniciada com *QueryID*. Tem de se chamar esta função para cada variável na instrução. O QMF para Windows não tenta combinar valores com variáveis, de forma que tem de chamar esta função pela mesma ordem das variáveis na instrução SQL.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeStaticQuery()</code> .
<i>Type</i>	O tipo de dados SQL do valor a transmitir ao servidor de base de dados. Este valor influencia a conversão de <i>Value</i> de um tipo de dados VARIANT para o valor realmente transmitido.
<i>Value</i>	O valor de dados a substituir na instrução. Para especificar um valor nulo, o tipo da variante deve ser definido como VT_EMPTY.

Os valores válidos para *Type* incluem:

Valor	Significado
384 (RSDT_DATE)	Data
388 (RSDT_TIME)	Hora
392 (RSDT_TIMESTAMP)	Marca de Hora
448 (RSDT_VARCHAR)	Cadeia de caracteres de comprimento variável
452 (RSDT_CHAR)	Cadeia de caracteres
464 (RSDT_VARGRAPHIC)	Gráfico de comprimento variável
468 (RSDT_GRAPHIC)	Gráfico
480 (RSDT_FLOAT)	Número de vírgula flutuante
496 (RSDT_INTEGER)	Número inteiro de 4 bytes
500 (RSDT_SMALLINT)	Número inteiro de 2 bytes

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()` para obter informações de erro adicionais.

BindDecimalHostVariable()

BindDecimalHostVariable curto (*CollectionName* BSTR, *PackageName* BSTR, *SectionNumber* curto, *Number* curto, *Name* BSTR, *DataType* curto, *Precision* curto, *Scale* curto)

Descrição

Esta função associa uma variável na secção especificada. Inclua o texto ":H" no texto SQL como marcador de posição para uma variável de sistema central. Para cada variável de sistema central decimal no texto SQL tem de chamar BindDecimalHostVariable() para especificar informações acerca da variável.

Parâmetros

Nome	Descrição
CollectionName	O ID de coleção do pacote que pretende associar.
PackageName	O nome do pacote que pretende associar.
SectionNumber	O número de secção da instrução dentro da coleção e do pacote que pretende associar.
Number	O identificador para a variável que pretende associar. A primeira variável na instrução SQL é a variável 0 e assim sucessivamente.
Name	Utilizado pelo servidor de base de dados para fins de diagnóstico. Este valor não é validado nem exigido pelo QMF para Windows.
DataType	O tipo de dados SQL da variável. O único valor válido para BindDecimalHostVariable() é 484 (RSDT_DECIMAL).
Precision	A precisão do valor decimal.
Scale	A escala do valor decimal.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType(), GetLastErrorSQLCode(), GetLastErrorSQLError(), ou GetLastErrorSQLState() para obter informações de erro adicionais.

BindHostVariable()

BindHostVariable curto (*CollectionName* BSTR, *PackageName* BSTR, *SectionNumber* curto, *Number* curto, *Name* BSTR, *DataType* curto, *Length* curto)

Descrição

Esta função associa uma variável na secção especificada. Inclua o texto ":H" no texto SQL como marcador de posição para uma variável de sistema central. Para cada variável de sistema central no texto SQL tem de chamar BindHostVariable() para especificar informações acerca da variável.

Parâmetros

Nome	Descrição
<i>CollectionName</i>	O ID de coleção do pacote que pretende associar.
<i>PackageName</i>	O nome do pacote que pretende associar.
<i>SectionNumber</i>	O número de secção da instrução dentro da coleção e do pacote que pretende associar.
<i>Number</i>	O identificador para a variável que pretende associar. A primeira variável na instrução SQL é a variável 0 e assim sucessivamente.
<i>Name</i>	Utilizado pelo servidor de base de dados para fins de diagnóstico. Este valor não é validado nem exigido pelo QMF para Windows.
<i>DataType</i>	O tipo de dados SQL da variável.
<i>Length</i>	O comprimento da variável.

Os valores válidos para *DataType* incluem:

Valor	Significado
384 (RSDT_DATE)	Data
388 (RSDT_TIME)	Hora
392 (RSDT_TIMESTAMP)	Marca de Hora
448 (RSDT_VARCHAR)	Cadeia de caracteres de comprimento variável
452 (RSDT_CHAR)	Cadeia de caracteres
464 (RSDT_VARGRAPHIC)	Gráfico de comprimento variável
468 (RSDT_GRAPHIC)	Gráfico
480 (RSDT_FLOAT)	Número de vírgula flutuante
484 (RSDT_DECIMAL)	Decimal
496 (RSDT_INTEGER)	Número inteiro de 4 bytes
500 (RSDT_SMALLINT)	Número inteiro de 2 bytes

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

BindSection()

`BindSection` curto (*CollectionName* BSTR, *PackageName* BSTR, *SectionNumber* curto, *SQLText* BSTR)

Descrição

Esta função define o texto SQL a utilizar no número de secção especificado da colecção e do pacote durante a associação.

Parâmetros

Nome	Descrição
CollectionName	O ID de colecção do pacote que pretende associar.
PackageName	O nome do pacote que pretende associar.
SectionNumber	O número de secção da instrução dentro da colecção e do pacote que pretende associar.
SQLText	O texto SQL para a instrução que pretende associar.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

CancelBind()

CancelBind curto (*CollectionName* BSTR, *PackageName* BSTR)

Descrição

Esta função cancela uma operação de associação anteriormente iniciada. São libertadas todas as informações relativas ao pacote indicado.

Parâmetros

Nome	Descrição
CollectionName	O ID de colecção do pacote que pretende associar.
PackageName	O nome do pacote que pretende associar.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

ChangePassword()

ChangePassword curto(*NewPassword* BSTR)

Descrição

Esta função altera a palavra-passe para o ID de utilizador anteriormente especificado na chamada `InitializeServer()`.

Nota: Nem todos os tipos de servidores de base de dados suportam alteração de palavras-passe. Se o servidor especificado na chamada `InitializeServer()` não suportar alteração de palavras-passe será devolvido um erro e a palavra-passe não será alterada.

Parâmetros

Nome	Descrição
NewPassword	A nova palavra-passe.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

ClearList()

ClearList curto(*Type* curto)

Descrição

Esta função reinicia a lista interna especificada pelo parâmetro *Type*.

Parâmetros

Nome	Descrição
<i>Type</i>	O valor RSL_SERVER ou RSL_QUERY.

Valor de Retorno

Zero se tiver êxito, RS_ERROR_OUTOFRANGE se não tiver êxito.

Tópicos Relacionados

Open()

Close()

Close curto(*QueryID* longo)

Descrição

Esta função fecha uma consulta e invalida *QueryID*. Se existir um cursor aberto para a consulta, este será fechado libertando assim a base de dados para outros utilizadores. Esta função não termina a ligação ao servidor de base de dados. Visto que a ligação permanece aberta, não será efectuada remoção de alterações nem consolidação.

Nota: O nome desta função entra em conflito com a palavra-chave Close do Microsoft Access 2.0. Se utilizar MS Access 2.0, coloque parêntesis rectos [] em redor do nome da função.

Parâmetros

Nome	Descrição
QueryID	O ID da consulta tal como é devolvido por InitializeQuery().

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType(), GetLastSQLCode(), GetLastSQLError(), ou GetLastSQLState() para obter informações de erro adicionais.

Tópicos Relacionados

Execute()

Open()

Commit()

Close curto(*QueryID* longo)

Descrição

Esta função consolida quaisquer alterações efectuadas na unidade de trabalho actual, finaliza esta última, fecha cursores abertos e invalida todos os IDs de Consulta pendentes.

Nota: O nome desta função entra em conflito com a palavra-chave Commit do Microsoft Access 2.0. Se utilizar MS Access 2.0, coloque parêntesis rectos [] em redor do nome da função.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType(), GetLastSQLCode(), GetLastSQLError(), ou GetLastSQLState() para obter informações de erro adicionais.

Tópicos Relacionados

Rollback()

CompleteQuery()

CompleteQuery curto(*QueryID* longo)

Descrição

Esta função busca todas as linhas de um conjunto de resultados e armazena-as internamente no QMF para Windows. Se existir um cursor aberto para a consulta, este será fechado libertando assim a base de dados para outros utilizadores. Pode-se utilizar FetchNextRow() ou FetchNextRows() para obter as linhas. Chame Close() quando concluir esta consulta.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

CopyToClipboard()

`CopyToClipboard` curto(*QueryID* longo, *FirstRow* longo, *FirstCol* longo, *LastRow* longo, *LastCol* longo, *IncludeColHeadings* BOOL, [*DateTimeFormat* VARIANT])

Descrição

Esta função copia o intervalo especificado de colunas e linhas para a Área de Transferência. Se não tiver obtido dados de linha para todas as linhas que pretende copiar para a Área de Transferência, chame `CompleteQuery()` antes de chamar esta função. Será devolvida uma mensagem de erro se tentar copiar linhas que não tenham sido obtidas da base de dados.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .
<i>FirstRow</i>	A primeira linha que pretende incluir na cópia.
<i>FirstCol</i>	A primeira coluna que pretende incluir na cópia.
<i>LastRow</i>	A última linha que pretende incluir na cópia ou -1 se estiverem incluídas todas as linhas.
<i>LastCol</i>	A última coluna que pretende incluir na cópia ou -1 se estiverem incluídas todas as colunas.
<i>IncludeColHeadings</i>	Utilize um valor diferente de zero para incluir os títulos de coluna na primeira linha e zero para não os incluir.
<i>DateTimeFormat</i>	O formato, a título opcional, a utilizar para valores de data e hora. Os valores válidos são 0 (formato ISO), 1 (formato USA), 2 (formato EUR), 3 (formato JIS), ou 4 (formato do Painel de Controlo do Windows). O valor predefinido é 4.

Nota: O valor da primeira linha num conjunto de resultados é 0 e o valor da última linha é um número abaixo do número total de linhas. O valor da

primeira coluna num conjunto de resultados é 0 e o valor da última coluna é um número abaixo do número total de colunas.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()` para obter informações de erro adicionais. Se o conjunto de resultados estiver vazio, ou se não tiverem sido obtidas linhas da base de dados, será devolvido um valor diferente de zero, a menos que *FirstRow*=0 e *LastRow*=1. Neste caso, será devolvido zero e será copiada uma cadeia vazia para a Área de Transferência.

DeleteQMFObject()

DeleteQMFObject curto(*OwnerAndName* BSTR)

Descrição

Esta função elimina um objecto do QMF (consulta, formulário, procedimento ou tabela).

Parâmetros

Nome	Descrição
<i>OwnerAndName</i>	Uma cadeia contendo o proprietário e o nome, separados por um ponto, do objecto que pretende eliminar. Por exemplo, João.Consulta2

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

EndBind()

EndBind curto(*CollectionName* BSTR, *PackageName* BSTR)

Descrição

Esta função conclui o processo de associação relativamente a um pacote SQL estático. Ao chamar esta função fará com que o QMF para Windows envie a totalidade das informações acerca do actual pacote à base de dados para processamento.

Parâmetros

Nome	Descrição
<i>CollectionName</i>	O nome da colecção utilizado na anterior chamada <code>StartBind()</code> .
<i>PackageName</i>	O nome do pacote utilizado na anterior chamada <code>StartBind()</code> .

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

Execute()

Execute curto(*QueryID* longo)

Descrição

Esta função executa uma instrução SQL que utiliza um verbo SQL diferente de SELECT. Utilize `Execute()` quando a instrução não devolver resultados. Para instruções que devolvam resultados, utilize `ExecuteEx()`. Para instruções que utilizem o verbo SELECT, utilize `Open()` em vez de `Execute()` ou `ExecuteEx()`. Para saber qual o verbo usado por uma consulta, chame `GetQueryVerb()`.

Nota: O nome desta função entra em conflito com a palavra-chave `Execute` do Microsoft Access 2.0. Se utilizar MS Access 2.0, coloque parêntesis rectos [] em redor do nome da função.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

Tópicos Relacionados

`Execute()`

`Open()`

ExecuteEx()

ExecuteEx curto(*QueryID*, VARIANT* *Result* longo)

Descrição

Esta função executa uma instrução SQL que utiliza um verbo SQL diferente de SELECT. Utilize `ExecuteEx()` quando a instrução devolver resultados, por exemplo, com a instrução `SELECT INTO`. Para instruções que não devolvam resultados, utilize `Execute()`. Para instruções que utilizem o verbo SELECT, utilize `Open()` em vez de `Execute()` ou `ExecuteEx()`. Para saber qual o verbo usado por uma consulta, chame `GetQueryVerb()`.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .
<i>Result</i>	<p>Um apontador para uma VARIANT na qual o resultado está armazenado. O resultado será uma matriz (tipo de variante VT_ARRAY VT_VARIANT) contendo um valor para cada coluna no resultado.</p> <p>Cada valor estará especificado no seu tipo de dados nativo ou no tipo de dados da variante mais próxima. Os tipos de retorno suportados são: cadeia (tipo variante VT_BSTR), flutuante (tipo variante VT_R4), duplo (tipo variante VT_R8), curto (tipo variante VT_I2), longo (tipo variante VT_I4) e binário (tipo variante VT_UI1 VT_ARRAY).</p> <p>Tem de iniciar devidamente a VARIANT antes de chamar esta função. O Visual Basic faz isto automaticamente. Os programadores de Visual C++ têm de chamar <code>VariantInit()</code>.</p>

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

ExecuteStoredProcedure()

`ExecuteStoredProcedure` curto(*QueryID*, longo [VARIANT *vaCommitOK*], [VARIANT *vaMaxResultSets*], [VARIANT *vaColumnNames*], [VARIANT *vaColumnLabels*], [VARIANT *vaColumnComments*])

Descrição

Esta função executa uma instrução SQL que utiliza o verbo SQL CALL para executar um procedimento armazenado no servidor de base de dados. Use `ExecuteStoredProcedure()` quando o procedimento armazenado não devolver resultados nenhuns (em vez ou além de conjuntos de resultados). No caso de procedimentos armazenados que devolvam de facto resultados, utilize `ExecuteStoredProcedureEx()`.

Para iniciar um procedimento armazenado para execução com `ExecuteStoredProcedure()`, chame primeiro `InitializeQuery()` especificando uma instrução SQL que utilize a instrução CALL. O procedimento armazenado tem de ser especificado como literal na instrução CALL. Quaisquer parâmetros especificados na instrução CALL (constantes ou outros) serão ignorados. Em vez disso, utilize `AddHostVariable()` para especificar as variáveis de input e output.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> . O texto SQL para a consulta deve especificar uma instrução <code>CALL</code> .
<i>vaCommitOK</i>	Um valor Booleano opcional especificando se o procedimento armazenado pode, ou não, consolidar a unidade de trabalho ou se esta operação deve ser restrita. O valor predefinido é <code>True</code> .
<i>vaMaxResultSets</i>	Um valor numérico opcional especificando o número máximo de conjuntos de resultados que se deve permitir que o procedimento armazenado devolva. Especifique zero se não quiser que o procedimento armazenado devolva conjuntos de resultados ou se o servidor de base de dados não suportar a devolução de conjuntos de resultados por parte de procedimentos armazenados através de DRDA.
<i>vaColumnNames</i>	Um valor Booleano opcional especificando se a base de dados deve, ou não, devolver nomes de coluna em cada conjunto de resultados devolvido.
<i>vaColumnLabels</i>	Um valor Booleano opcional especificando se a base de dados deve, ou não, devolver identificadores de coluna em cada conjunto de resultados devolvido.
<i>vaColumnComments</i>	Um valor Booleano opcional especificando se a base de dados deve, ou não, devolver comentários de coluna em cada conjunto de resultados devolvido.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

ExecuteStored ProcedureEx()

`ExecuteStoredProcedureEx` curto(*QueryID*, *VARIANT** *Result*, longo[*VARIANT vaCommitOK*], [*VARIANT vaMaxResultSets*], [*VARIANT vaColumnNames*], [*VARIANT vaColumnLabels*], [*VARIANT vaColumnComments*])

Descrição

Esta função executa uma instrução SQL que utiliza o verbo SQL `CALL` para executar um procedimento armazenado no servidor de base de dados. Use `ExecuteStoredProcedureEx()` quando o procedimento armazenado devolver resultados (em vez ou além de conjuntos de resultados). No caso de procedimentos armazenados que devolvam de facto resultados, utilize `ExecuteStoredProcedureEx()`.

Para iniciar um procedimento armazenado para execução com `ExecuteStoredProcedure()`, chame primeiro `InitializeQuery()` especificando uma instrução SQL que utilize a instrução `CALL`. O procedimento armazenado tem de ser especificado como literal na instrução `CALL`. Quaisquer parâmetros especificados na instrução `CALL` (constantes ou outros) serão ignorados. Em vez disso, utilize `AddHostVariable()` para especificar as variáveis de input e output.

Se o procedimento armazenado devolver conjuntos de resultados, chame `GetStoredProcedureResultSets()` para obter os IDs de consulta relativos aos conjuntos de resultados.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> . O texto SQL para a consulta deve especificar uma instrução <code>CALL</code> .
<i>Result</i>	Um apontador para uma <code>VARIANT</code> na qual o resultado está armazenado. O resultado será uma matriz (tipo de variante <code>VT_ARRAY</code> <code>VT_VARIANT</code>) contendo um valor para cada coluna no resultado. Cada valor estará especificado no seu tipo de dados nativo ou no tipo de dados da variante mais próxima. Os tipos de retorno suportados são: cadeia (tipo variante <code>VT_BSTR</code>), flutuante (tipo variante <code>VT_R4</code>), duplo (tipo variante <code>VT_R8</code>), curto (tipo variante <code>VT_I2</code>), longo (tipo variante <code>VT_I4</code>) e binário (tipo variante <code>VT_UI1</code> <code>VT_ARRAY</code>). Tem de iniciar devidamente a <code>VARIANT</code> antes de chamar esta função. O Visual Basic faz isto automaticamente. Os programadores de Visual C++ têm de chamar <code>VARIANTInit()</code> .
<i>vaCommitOK</i>	Um valor Booleano opcional especificando se o procedimento armazenado pode, ou não, consolidar a unidade de trabalho ou se esta operação deve ser restrita. O valor predefinido é <code>True</code> .
<i>vaMaxResultSets</i>	Um valor numérico opcional especificando o número máximo de conjuntos de resultados que se deve permitir que o procedimento armazenado devolva. Especifique zero se não quiser que o procedimento armazenado devolva conjuntos de resultados ou se o servidor de base de dados não suportar a devolução de conjuntos de resultados por parte de procedimentos armazenados através de <code>DRDA</code> .
<i>vaColumnNames</i>	Um valor Booleano opcional especificando se a base de dados deve, ou não, devolver nomes de coluna em cada conjunto de resultados devolvido.

<i>vaColumnLabels</i>	Um valor Booleano opcional especificando se a base de dados deve, ou não, devolver identificadores de coluna em cada conjunto de resultados devolvido.
<i>vaColumnComments</i>	Um valor Booleano opcional especificando se a base de dados deve, ou não, devolver comentários de coluna em cada conjunto de resultados devolvido.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

Export()

Export curto(*QueryID* longo, *FirstRow* longo, *FirstCol* longo, *LastRow* longo, *LastCol* longo, *Format* curto, *StringDelimiter* curto, *ColumnDelimiter* curto, *IncludeColHeadings* BOOL, *FileName* BSTR, [VARIANT *DateTimeFormat*])

Descrição

Esta função copia o intervalo especificado de colunas e linhas para a Área de Transferência. Se não tiver obtido dados de linha para todas as linhas que pretende copiar para a Área de Transferência, chame `CompleteQuery()` antes de chamar esta função. Será devolvida uma mensagem de erro se tentar copiar linhas que não tenham sido obtidas da base de dados.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .
<i>FirstRow</i>	A primeira linha que pretende incluir na exportação.
<i>FirstCol</i>	A primeira coluna que pretende incluir na exportação.
<i>LastRow</i>	A última linha que pretende incluir na cópia ou -1 se estiverem incluídas todas as linhas.
<i>LastCol</i>	A última coluna que pretende incluir na cópia ou -1 se estiverem incluídas todas as colunas.
<i>IncludeColHeadings</i>	Utilize um valor diferente de zero para incluir os títulos de coluna na primeira linha e zero para não os incluir.
<i>Filename</i>	Uma cadeia contendo o nome do ficheiro para o qual pretende gravar a exportação.
<i>DateTimeFormat</i>	O formato, a título opcional, a utilizar para valores de data e hora. Os valores válidos são 0 (formato ISO), 1 (formato USA), 2 (formato EUR), 3 (formato JIS), ou 4 (formato do Painel de Controlo do Windows). O valor predefinido é 4.

Nota: O valor da primeira linha num conjunto de resultados é 0 e o valor da última linha é um número abaixo do número total de linhas. O valor da primeira coluna num conjunto de resultados é 0 e o valor da última coluna é um número abaixo do número total de colunas.

Nome	Descrição
<i>Format</i>	Especifica o formato de output.

Valor	Significado
0 (RSEF_TEXT)	O ficheiro de output é escrito em formato de texto normal.
1 (RSEF_HTML)	O ficheiro de output é escrito em formato HTML e os dados são organizados numa tabela HTML.
2 (RSEF_CSV)	O ficheiro de output é escrito em CSV (valores separados por vírgulas).
3 (RSEF_PCIXF)	O ficheiro de output é escrito em formato PC/IXF.
4 (RSEF_S370IXF)	O ficheiro de output é escrito em formato System/370 IXF.

Nome	Descrição
<i>String Delimiter</i>	Especifica o delimitador de cadeia. Este parâmetro é ignorado se <i>Format</i> for RSEF_HTML.

Valor	Significado
0 (RSSD_NONE)	Não é utilizado nenhum delimitador de cadeia.
1 (RSSD_SINGLEQUOTE)	As cadeias são delimitadas por uma única aspa (').
2 (RSSD_DOUBLEQUOTE)	As cadeias são delimitadas por aspas duplas (").

Nome	Descrição
<i>Column Delimiter</i>	Especifica o delimitador de coluna. Este parâmetro é ignorado se <i>Format</i> for RSEF_HTML.

Valor	Significado
0 (RSCD_SPACE)	As colunas são delimitadas por um espaço ().
1 (RSCD_TAB)	As colunas são delimitadas por uma marca de tabulação (\t).
2 (RSCD_COMMA)	As colunas são delimitadas por uma vírgula (,).

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()` para obter informações de erro adicionais. Se o conjunto de resultados estiver vazio, ou se não tiverem sido obtidas linhas da base de dados, será devolvido um valor diferente de zero, a menos que *FirstRow=0* e *LastRow=1*. Neste caso, será devolvido zero e será escrito um ficheiro vazio.

Tópicos Relacionados

`CopyToClipboard()`

ExportForm()

`ExportForm` curto(*OwnerAndName* BSTR, *FileName* BSTR)

Descrição

Esta função exporta o formulário QMF especificado para o ficheiro designado.,

Parâmetros

Nome	Descrição
<i>OwnerAndName</i>	Uma cadeia contendo o proprietário e o nome, separados por um ponto, do formulário que pretende exportar. Por exemplo, João.Consulta2
<i>FileName</i>	Uma cadeia contendo o nome do ficheiro para o qual pretende gravar o formulário exportado.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

Tópicos Relacionados

`PrintReport()`

ExportReport()

`ExportReport` curto(*QueryID* longo, *SourceType* curto, *Source* BSTR, *OutputFileName* BSTR, *PageLength* curto, *PageWidth* curto, *IncludeDateTime* BOOL, *IncludePageNumbers* BOOL, [VARIANT *Format*], [VARIANT *UseFormPageSetup*])

Descrição

Esta função cria um relatório para a consulta especificada e grava-o num ficheiro. Especifica-se o esquema e a formatação do relatório num formulário QMF. O ficheiro de output é um ficheiro de texto ASCII com cada linha

separada por um par de caracteres de retorno do carro e mudança de linha e cada página separada por um carácter de fim de página. Recomenda-se ver o ficheiro de output com um tipo de letra de caracteres fixos.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .
<i>Source</i>	O nome (nome de ficheiro ou <code>Proprietário.Nome</code>) do formulário que pretende usar.
<i>OutputFileName</i>	O nome do ficheiro para o qual pretende gravar o relatório.
<i>PageLength</i>	O número de linhas em cada página do relatório. Um <i>PageLength</i> de -1 especifica output contínuo (sem quebras de página a menos que o relatório seja mais largo do que a <i>PageWidth</i>).
<i>IncludeDateTime</i>	Um valor diferente de zero especifica que a data e a hora estão incluídas no final de cada página. Um valor zero especifica que a data e a hora não estão incluídas.
<i>IncludePageNumbers</i>	Um valor diferente de zero especifica que os números de página estão incluídos no final de cada página. Um valor zero especifica que os números de página não estão incluídos.
<i>Format</i>	Especifica, a título opcional, o formato do relatório exportado. Se for zero, o formato é texto normal, especificando que o output deve ser exactamente o produzido pelo formulário (texto ou HTML, consoante o tipo de formulário). Se for diferente de zero, o formato é HTML, especificando que o output deve ser HTML. No caso de formulários diferentes de HTML, o output é convertido em HTML adicionando identificadores HTML no início e no fim do output. O valor predefinido é zero.
<i>DateTimeFormat</i>	O formato, a título opcional, a utilizar para valores de data e hora. Os valores válidos são 0 (formato ISO), 1 (formato USA), 2 (formato EUR), 3 (formato JIS), ou 4 (formato do Painel de Controlo do Windows). O valor predefinido é 4.
<i>Format</i>	O formato do ficheiro de output.
<i>UseFormPageSetup</i>	A título opcional, se o valor diferente de zero especificar que os parâmetros <i>PageLength</i> , <i>PageWidth</i> , <i>IncludeDateTime</i> e <i>IncludePageNumbers</i> devem ser ignorados, e os respectivos valores devem ser tomados daqueles guardados no formulário especificado. O valor predefinido é zero.
Valor	Significado
0 (RSF_DEFAULT)	Utilize o formulário predefinido. <i>FormName</i> deveria ser uma cadeia vazia.

1 (RSF_DATABASE)	Utilize um formulário da base de dados. Especifique o proprietário e o nome do formulário (Proprietário.Nome) no parâmetro <i>FormName</i> . Para usar um formulário localizado num servidor de base de dados diferente, utilize primeiro <i>ExportForm()</i> para exportar o formulário para um ficheiro e em seguida especifique <i>SourceType</i> de RSF_FILE.
2 (RSF_FILE)	Utilize um formulário contido num ficheiro. Especifique o nome do ficheiro no parâmetro <i>FormName</i> .

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar *GetLastErrorString()* ou *GetLastErrorType()*, *GetLastSQLCode()*, *GetLastSQLError()*, ou *GetLastSQLState()* para obter informações de erro adicionais.

Tópicos Relacionados

ExportForm()

FastSaveData()

FastSaveData curto(*QueryID* longo, *Replace* BOOL, *Tablename* BSTR, *TableSpaceName* BSTR, [VARIANT *Comment*])

Descrição

Esta função cria um relatório para a consulta especificada e grava-o num ficheiro. Especifica-se o esquema e a formatação do relatório num formulário QMF. O ficheiro de output é um ficheiro de texto ASCII com cada linha separada por um par de caracteres de retorno do carro e mudança de linha e cada página separada por um carácter de fim de página. Recomenda-se ver o ficheiro de output com um tipo de letra de caracteres fixos.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <i>InitializeQuery()</i> .
<i>Replace</i>	Utilize um valor diferente de zero se preferir que os dados especificados substituam dados existentes na tabela. Utilize zero se preferir que os dados especificados sejam anexados a dados existentes na tabela.
<i>TableName</i>	O nome da tabela onde pretende armazenar os dados. Se a tabela não existir, o QMF para Windows irá criá-la.

TableName	O nome do espaço de tabela onde esta existe ou será criada. Se <i>TableName</i> for NULL ou uma cadeia vazia, o QMF para Windows utiliza o espaço de tabela predefinido. Se tiver configurado o QMF para Windows para utilizar sempre o espaço de tabela predefinido, este parâmetro será ignorado. Consulte RSR_SDDIFFERENTTS na descrição para GetResourceLimit().
Comment	A título opcional, uma cadeia que especifica um comentário para a tabela na qual são guardados os dados.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType(), GetLastErrorSQLCode(), GetLastErrorSQLError(), ou GetLastErrorSQLState() para obter informações de erro adicionais.

FetchNextRow()

FetchNextRow curto(*QueryID* longo, VARIANT* *Row*)

Descrição

Esta função busca a linha de dados seguinte da base de dados.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por InitializeQuery().
Row	Um apontador para uma VARIANT na qual o resultado está armazenado. O resultado será uma matriz (tipo de variante VT_ARRAY VT_VARIANT) contendo um valor para cada coluna no resultado. Chame GetColumnCount() para determinar o número de valores na matriz. Cada valor estará especificado no seu tipo de dados nativo ou no tipo de dados da variante mais próxima. Os tipos de retorno suportados são: cadeia (tipo variante VT_BSTR), flutuante (tipo variante VT_R4), duplo (tipo variante VT_R8), curto (tipo variante VT_I2), longo (tipo variante VT_I4) e binário (tipo variante VT_UI1 VT_ARRAY). Ao atingir o final do conjunto de resultados (não há mais linhas a buscar) ou se o conjunto de resultados estiver vazio, o resultado será vazio (tipo de variante VT_EMPTY) em vez de uma matriz . Tem de iniciar devidamente a VARIANT antes de chamar esta função. O Visual Basic faz isto automaticamente. Os programadores de Visual C++ têm de chamar VariantInit().

Nota: Devido a um erro no Microsoft Excel 7.0 e no Microsoft Access 7.0 (e possivelmente em outros produtos de 32 bits da Microsoft que utilizam o Visual Basic for Applications), dados de cadeia em variáveis Variant recebidos do QMF para Windows podem não ser convertidos de Unicode (usado por OLE) em ANSI (usado por VBA). Nesta situação, só é apresentado o primeiro carácter da cadeia. Para resolver este problema, defina a variável igual a uma cadeia vazia antes de chamar a função do QMF para Windows que utiliza a variável.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Ao atingir o final do conjunto de resultados, o valor de retorno é -1. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

Tópicos Relacionados

`FetchNextRows()`

FetchNextRowEx()

`FetchNextRowEx` curto(*QueryID* longo)

Descrição

Esta função busca a linha de dados seguinte da base de dados. Pode usar esta função em ambientes que não suportem matrizes VARIANT, tais como o Microsoft Access 2.0. Use esta função em conjunto com `GetColumnValue()` para obter os dados em cada coluna para a linha actual.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Ao atingir o final do conjunto de resultados, o valor de retorno é -1. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

Tópicos Relacionados

`FetchNextRowsEx()`

FetchNextRows()

`FetchNextRows` curto(*QueryID* longo, VARIANT* *Rows*, *NumRows* longo*)

Descrição

Esta função busca o *NumRows* de dados seguinte da base de dados.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .
Row	<p>Um apontador para uma VARIANT na qual o resultado está armazenado. O resultado será uma matriz bidimensional (tipo de variante VT_ARRAY VT_VARIANT) contendo um valor para cada coluna em cada linha. Chame <code>GetColumnCount()</code> para determinar o número de colunas na matriz. As dimensões da matriz são [<i>NumRows</i>][<i>ColumnCount</i>], mesmo que o número de linhas não buscadas no conjunto de resultados seja menor que <i>NumRows</i> (neste caso, a matriz contém entradas extra e não usadas).</p> <p>Cada valor estará especificado no seu tipo de dados nativo ou no tipo de dados da variante mais próxima. Os tipos de retorno suportados são: cadeia (tipo variante VT_BSTR), flutuante (tipo variante VT_R4), duplo (tipo variante VT_R8), curto (tipo variante VT_I2), longo (tipo variante VT_I4) e binário (tipo variante VT_UI1 VT_ARRAY).</p> <p>Ao atingir o final do conjunto de resultados (não há mais linhas a buscar) ou se o conjunto de resultados estiver vazio, o resultado será vazio (tipo de variante VT_EMPTY) em vez de uma matriz</p> <p>Tem de iniciar devidamente a VARIANT antes de chamar esta função. O Visual Basic faz isto automaticamente. Os programadores de Visual C++ têm de chamar <code>VariantInit()</code>.</p>
<i>NumRows</i>	Um apontador para um longo contendo o número da linha a buscar. Se o número de linhas não buscadas no conjunto de resultados for menor que <i>NumRows</i> , <i>NumRows</i> é reposto para o número real de linhas contidas no resultado.

Nota: Devido a um erro no Microsoft Excel 7.0 e no Microsoft Access 7.0 (e possivelmente em outros produtos de 32 bits da Microsoft que utilizam o Visual Basic for Applications), dados de cadeia em variáveis Variant recebidos do QMF para Windows podem não ser convertidos de Unicode (usado por OLE) em ANSI (usado por VBA). Nesta situação, só é apresentado o primeiro carácter da cadeia. Para resolver este problema, defina a variável igual a uma cadeia vazia antes de chamar a função do QMF para Windows que utiliza a variável.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Ao atingir o final do conjunto de resultados, o valor de retorno é -1. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

Tópicos Relacionados

`FetchNextRow()`

FetchNextRowsEx()

`FetchNextRowsEx` curto(*QueryID* longo, *NumRows* longo*)

Descrição

Esta função busca o *NumRows* de dados seguinte da base de dados. Pode usar esta função em ambientes que não suportem matrizes VARIANT, tais como o Microsoft Access 2.0. Use esta função em conjunto com `GetColumnValueEx()` para obter os dados em cada coluna para uma dada linha.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .
<i>NumRows</i>	Um apontador para um longo contendo o número da linha a buscar. Se o número de linhas não buscadas no conjunto de resultados for menor que <i>NumRows</i> , <i>NumRows</i> é repostado para o número real de linhas contidas no resultado.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Ao atingir o final do conjunto de resultados, o valor de retorno é -1. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

Tópicos Relacionados

`FetchNextRowEx()`

FlushQMFCache()

`FlushQMFCache()` vazio

Descrição

Esta função diz ao QMF para Windows para limpar a sua memória cache de informações do QMF, descartando o conteúdo. Da próxima vez que o QMF para Windows necessitar de informações de QMF, irá obtê-las da base de dados. Regra geral, o QMF para Windows coloca em memória cache informações do QMF obtidas da base de dados para reduzir o tráfego desta

última e aumentar o rendimento. Chama-se esta função antes de chamar `GetQMFOBJECTInfo()`, `GetQMFOBJECTQueryText()`, ou `GetQMFOBJECTList()` para assegurar que as informações devolvidas estão actualizadas.

Valor de Retorno

Nenhum.

GetColumnCount()

`GetColumnCount` longo(*QueryID* longo)

Descrição

Esta função devolve o número de colunas no conjunto de resultados.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .

Valor de Retorno

O número de colunas em cada linha se for bem sucedido. Caso contrário, 0 ou -1. Se o valor de retorno for 0 ou -1, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

GetColumnDataValue()

`GetColumnDataValue` curto (*QueryID* longo, *Index* longo)

Descrição

Esta função devolve o valor de dados para a coluna especificada em *Index* relativo à linha de dados actual. Depois de chamar esta função, a propriedade *Value* pode ser interrogada acerca do valor devolvido. Utiliza-se esta função com `FetchNextRowEx()` para aceder aos dados numa única linha de dados.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .
<i>Index</i>	O índice com base zero da linha de dados a obter.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

GetColumnHeader()

GetColumnHeader BSTR(*QueryID* longo, *Index* longo, *Result* curto*)

Descrição

Esta função devolve o cabeçalho da coluna (nome da coluna) associado ao índice *Index*.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por InitializeQuery().
<i>Index</i>	O índice com base zero da linha de dados a obter.
<i>Result</i>	Zero se tiver êxito, diferente de zero se não tiver êxito. Se <i>Result</i> for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType(), GetLastErrorCode(), GetLastErrorError(), ou GetLastErrorSQLState() para obter informações de erro adicionais.

Nota: Os títulos de coluna não estão disponíveis para instruções SQL estáticas. No caso de IDs de consulta devolvidos por InitializeStaticQuery(), GetColumnHeader devolve uma cadeia do formulário "Coln" em que "n" é o número de coluna.

Valor de Retorno

A cadeia devolvida representa o nome da coluna tal como especificado no parâmetro *Index*.

GetColumnHeaderEx()

GetColumnHeaderEx curto(*QueryID* longo, *Index* longo)

Descrição

Esta função devolve o cabeçalho da coluna (nome da coluna) associado ao índice *Index*. Depois de chamar esta função, a propriedade *Value* pode ser interrogada acerca do valor devolvido.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por InitializeQuery().
<i>Index</i>	O índice com base zero da linha de dados a obter.

Nota: Os títulos de coluna não estão disponíveis para instruções SQL estáticas. No caso de IDs de consulta devolvidos por

InitializeStaticQuery(), GetColumnHeader devolve uma cadeia do formulário "Coln" em que "n" é o número de coluna.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for zero, consulte a propriedade *Value* relativamente à cadeia que representa o nome da coluna. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType(), GetLastSQLCode(), GetLastSQLError(), ou GetLastSQLState() para obter informações de erro adicionais.

GetColumnHeadings()

GetColumnHeadings curto(*QueryID* longo, VARIANT* *Headings*)

Descrição

Esta função devolve os títulos de coluna (também referidos como nomes de coluna).

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por InitializeQuery().
<i>Headings</i>	Um apontador para uma VARIANT na qual o resultado está armazenado. O resultado será uma matriz de cadeias (tipo de variante VT_ARRAY VT_BSTR) contendo uma cadeia para cada título de coluna. Tem de iniciar devidamente a VARIANT antes de chamar esta função. O Visual Basic faz isto automaticamente. Os programadores de Visual C++ têm de chamar VariantInit().

Nota: Devido a um erro no Microsoft Excel 7.0 e no Microsoft Access 7.0 (e possivelmente em outros produtos de 32 bits da Microsoft que utilizam o Visual Basic for Applications), dados de cadeia em variáveis Variant recebidos do QMF para Windows podem não ser convertidos de Unicode (usado por OLE) em ANSI (usado por VBA). Nesta situação, só é apresentado o primeiro carácter da cadeia. Para resolver este problema, defina a variável igual a uma cadeia vazia antes de chamar a função do QMF para Windows que utiliza a variável.

Nota: Os títulos de coluna não estão disponíveis para instruções SQL estáticas. No caso de IDs de consulta devolvidos por InitializeStaticQuery(), GetColumnHeadings devolve as cadeias "Col1", "Col2", etc.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

GetColumnValue()

`GetColumnValue` curto(*QueryID* longo, *Index* longo, *VARIANT** *Value*)

Descrição

Esta função devolve o valor de dados para a coluna especificada em *Index* relativo à linha de dados actual. Utiliza-se esta função com `FetchNextRowEx()` para aceder aos dados numa única linha de dados.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .
<i>Index</i>	O índice com base zero da linha de dados a obter.
<i>Value</i>	Um apontador para uma <i>VARIANT</i> onde pretende armazenar os resultados. O resultado é o valor de dados baseado no tipo de variante. Tem de iniciar devidamente a <i>VARIANT</i> antes de chamar esta função. O Visual Basic faz isto automaticamente. Os programadores de Visual C++ têm de chamar <code>VariantInit()</code> .

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

GetColumnValueEx()

`GetColumnValueEx` curto(*QueryID* longo, *RowIndex* longo, *ColIndex* longo, *VARIANT** *Value*)

Descrição

Esta função devolve o valor de dados para a coluna especificada em *ColIndex* relativamente à linha de dados especificada em *RowIndex*. Utiliza-se esta função com `FetchNextRowsEx()` para aceder aos dados numa única linha de dados.

Parâmetros

Nome	Descrição
------	-----------

<i>QueryID</i>	O ID da consulta tal como é devolvido por InitializeQuery().
RowIndex	O índice com base zero da linha a obter.
ColIndex	O índice com base zero da coluna a obter.
Value	Um apontador para uma VARIANT onde pretende armazenar o resultado. Pode-se consultar a variante resultante para saber o tipo de dados para processamento adicional. Tem de iniciar devidamente a VARIANT antes de chamar esta função. O Visual Basic faz isto automaticamente. Os programadores de Visual C++ têm de chamar VariantInit().

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType(), GetLastErrorSQLCode(), GetLastErrorSQLError(), ou GetLastErrorSQLState() para obter informações de erro adicionais.

GetDefaultServerName()

GetDefaultServerName() BSTR

Descrição

Esta função devolve uma cadeia contendo o nome do servidor predefinido.

Valor de Retorno

Uma cadeia que especifica o nome de servidor predefinido.

GetGlobalVariable()

GetGlobalVariable BSTR(*Name* BSTR)

Descrição

Esta função obtém o valor da variável global especificada.

Parâmetros

Nome	Descrição
Name	Uma cadeia que contém o nome da variável que pretende definir.

Valor de Retorno

Uma cadeia contendo o valor da variável global, ou NULL se a variável não tiver valor ou se ocorrer um erro.

GetHostVariableNames()

GetHostVariableNames curto(*QueryID* longo, VARIANT* *Names*)

Descrição

Esta função devolve uma matriz dos nomes de todas as variáveis de sistema central referenciadas na consulta especificada. A consulta tem de ser estática e referenciar variáveis de sistema central (quer armazenadas com a consulta QMF quer criadas por `AddHostVariable()`).

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .
Names	Um apontador para uma VARIANT onde pretende armazenar a matriz de resultados.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` para obter informações de erro adicionais.

GetHostVariableTypes()

`GetHostVariableTypes` curto(*QueryID* longo, VARIANT* *Types*)

Descrição

Esta função devolve uma matriz dos tipos de dados de todas as variáveis de sistema central referenciadas na consulta especificada. A consulta tem de ser estática e referenciar variáveis de sistema central (quer armazenadas com a consulta QMF quer criadas por `AddHostVariable()`).

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .
Types	Um apontador para uma VARIANT onde pretende armazenar a matriz de resultados.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` para obter informações de erro adicionais.

GetLastErrorString()

`GetLastErrorString()`BSTR

Descrição

Esta função devolve uma cadeia contendo informações acerca do erro mais recente. Se chamar esta função depois de uma função que tenha executado

com êxito (sem erros), esta função devolverá informações acerca do último erro ocorrido durante uma anterior chamada de função. Para evitar confusões, chame sempre esta função imediatamente depois de chamar uma função que devolveu um erro.

Valor de Retorno

Uma cadeia contendo informações de erro. Se não tiverem ocorrido erros desde que criou o objecto de API do QMF, será devolvido NULL.

Tópicos Relacionados

`GetLastErrorType()`

`GetLastSQLCode()`

`GetLastSQLError()`

`GetLastSQLState()`

GetLastErrorType()

`GetLastErrorType()` curto

Descrição

Esta função devolve o tipo do erro mais recente. Se chamar esta função depois de uma função que tenha executado com êxito (sem erros), esta função devolverá informações acerca do último erro ocorrido durante uma anterior chamada de função. Para evitar confusões, chame sempre esta função imediatamente depois de chamar uma função que devolveu um erro.

Valor de Retorno

O valor de retorno indica o tipo de erro:

Valor	Significado
0 (RS_ERROR_NONE)	Não ocorreram erros desde que o objecto da API do QMF para Windows foi criado.
1 (RS_ERROR_SQL)	Ocorreu um erro de SQL. Se o erro ocorreu durante uma chamada de uma função que tenha <i>QueryID</i> como argumento, chame <i>Close()</i> para fechar essa consulta. Não é efectuada nenhuma remoção de alterações. Pode continuar a utilizar o objecto da API do QMF para Windows, ainda que possa deparar com erros adicionais.
2 (RS_ERROR_USER_CANCEL)	Um utilizador cancelou a operação, normalmente por fazer clique em Cancelar na janela activa. Esta acção faz com que o QMF para Windows execute uma remoção de alterações implícita (invalidando todos os IDs de consulta pendentes) e destrua a ligação à base de dados. Tem de chamar <i>InitializeServer()</i> ou <i>ReinitializeServer()</i> se pretende continuar.

3 (RS_ERROR_FATAL_GOV)	Ocorreu um erro fatal do regulador. Pode dar-se o caso de a API do QMF para Windows ter esgotado o tempo por se ter excedido o máximo de tempo inactivo permitido. Esta acção faz com que o QMF para Windows execute uma remoção de alterações implícita (invalidando todos os IDs de consulta pendentes) e destrua a ligação à base de dados. Tem de chamar InitializeServer() ou ReinitializeServer() se pretende continuar.
4 (RS_ERROR_NONFATAL_GOV)	Ocorreu um erro que não é fatal do regulador. Foi excedido o número máximo permitido de linhas a buscar, ou o verbo SQL não é permitido. Se o erro ocorreu durante uma chamada de função que tenha <i>QueryID</i> como argumento, chame Close() para fechar essa consulta. Não é efectuada qualquer remoção de alterações nem a ligação à base de dados é afectada, de modo de pode continuar a utilizar o objecto da API do QMF para Windows.
5 (RS_ERROR_OTHER)	Ocorreu um erro geral. Não é efectuada nenhuma remoção de alterações. Pode continuar a utilizar o objecto da API do QMF para Windows, ainda que possa deparar com erros adicionais.

Tópicos Relacionados

GetLastErrorString()
 GetLastSQLCode()
 GetLastSQLError()
 GetLastSQLState()

GetLastSQLCode()

GetLastSQLCode() longo

Descrição

Esta função devolve o código SQL relativo ao erro mais recente. Se chamar esta função depois de uma função que tenha executado com êxito (sem erros), esta função devolverá informações acerca do último erro ocorrido durante uma anterior chamada de função. Para evitar confusões, chame sempre esta função imediatamente depois de chamar uma função que devolveu um erro.

Valor de Retorno

Os códigos SQL relativos ao erro mais recente. Se não tiverem ocorrido erros desde que criou o objecto de API do QMF, ou se o erro mais recente não for um erro SQL, será devolvido zero.

Tópicos Relacionados

GetLastErrorString()

GetLastErrorType()

GetLastSQLError()

GetLastSQLState()

GetLastSQLError()

VARIANT GetLastError()

Descrição

Esta função devolve informações detalhadas de erro SQL relativas ao erro mais recente. Se chamar esta função depois de uma função que tenha executado com êxito (sem erros), esta função devolverá informações acerca do último erro ocorrido durante uma anterior chamada de função. Para evitar confusões, chame sempre esta função imediatamente depois de chamar uma função que devolveu um erro.

Valor de Retorno

Uma matriz (tipo de variante VT_ARRAY | VT_VARIANT) contendo informações de erro. Se não tiverem ocorrido erros desde que criou o objecto de API do QMF, ou se o erro mais recente não for um erro SQL, será devolvido vazio (tipo de variante VT_EMPTY). A matriz tem o seguinte formato:

Elemento	Tipo	Conteúdo
0	longo (VT_I4)	Código
1	cadeia (VT_BSTR)	Estado
2	cadeia (VT_BSTR)	ErrProc
3	cadeia (VT_BSTR)	RDBName
4	longo (VT_I4)	ErrD1
5	longo (VT_I4)	ErrD2
6	longo (VT_I4)	ErrD3
7	longo (VT_I4)	ErrD4
8	longo (VT_I4)	ErrD5
9	longo (VT_I4)	ErrD6
10	cadeia (VT_BSTR)	Warn0
11	cadeia (VT_BSTR)	Warn1
12	cadeia (VT_BSTR)	Warn2
13	cadeia (VT_BSTR)	Warn3
14	cadeia (VT_BSTR)	Warn4
15	cadeia (VT_BSTR)	Warn5

16	cadeia (VT_BSTR)	Warn6
17	cadeia (VT_BSTR)	Warn7
18	cadeia (VT_BSTR)	Warn8
19	cadeia (VT_BSTR)	Warn9
20	cadeia (VT_BSTR)	WarnA
21	cadeia (VT_BSTR)	MessageTokens

Tópicos Relacionados

[GetLastErrorString\(\)](#)
[GetLastErrorType\(\)](#)
[GetLastSQLCode\(\)](#)
[GetLastSQLState\(\)](#)

GetLastSQLState()

GetLastSQLState() BSTR

Descrição

Esta função devolve o estado SQL relativo ao erro mais recente. Se chamar esta função depois de uma função que tenha executado com êxito (sem erros), esta função devolverá informações acerca do último erro ocorrido durante uma anterior chamada de função. Para evitar confusões, chame sempre esta função imediatamente depois de chamar uma função que devolveu um erro.

Valor de Retorno

Uma cadeia contendo o código SQL relativo ao erro mais recente. Se não tiverem ocorrido erros desde que criou o objecto de API do QMF, ou se o erro mais recente não for um erro SQL, será devolvido NULL.

Tópicos Relacionados

[GetLastErrorString\(\)](#)
[GetLastErrorType\(\)](#)
[GetLastSQLCode\(\)](#)
[GetLastSQLError\(\)](#)

GetOption()

GetOption curto(*Option* curto, VARIANT* *Value*)

Descrição

Obtém o valor da opção especificada no QMF para Windows.

Parâmetros

Nome	Descrição
<i>Opção</i>	Especifica a opção a obter.

Valor	Significado
0 (RSO_SERVER_DEFINITION_FILE)	Nome de ficheiro de definição de servidores.
1 (RSO_CPIC_DLL)	Nome de ficheiro da DLL fornecedora de CPI-C.
2 (RSO_CPIC_TIMEOUT_WARNING)	Tempo de espera de aviso CPI-C (em segundos). Este limite não é usado pela API do QMF para Windows.
3 (RSO_CPIC_TIMEOUT_CANCEL)	Tempo de espera de cancelamento CPI-C (em segundos).
4 (RSO_TCP_TIMEOUT_WARNING)	Tempo de espera de aviso TCP (em segundos). Este limite não é usado pela API do QMF para Windows.
5 (RSO_TCP_TIMEOUT_CANCEL)	Tempo de espera de cancelamento TCP (em segundos).
6 (RSO_DISPLAY_NULLS_STRING)	A cadeia a utilizar para apresentar valores nulos.
7 (RSO_ENTER_NULLS_STRING)	A cadeia a utilizar para inserir valores nulos.
8 (RSO_ENTER_DEFAULTS_STRING)	A cadeia a utilizar para inserir valores predefinidos.
9 (RSO_TRACE_FILE_1)	Nome do ficheiro de rastreio 1.
10 (RSO_TRACE_FILE_2)	Nome do ficheiro de rastreio 2.
11 (RSO_TCP_TRACE_LEVEL)	Nível de rastreio TCP.
12 (RSO_CPIC_TRACE_LEVEL)	Nível de rastreio CPI-C.
13 (RSO_DDM_TRACE_LEVEL)	Nível de rastreio DDM.
Value	Um apontador para uma VARIANT na qual o resultado está armazenado. O resultado será uma matriz (tipo de variante VT_ARRAY VT_VARIANT) contendo um valor para cada coluna no resultado. Chame GetColumnCount() para determinar o número de valores na matriz. Tem de iniciar devidamente a VARIANT antes de chamar esta função. O Visual Basic faz isto automaticamente. Os programadores de Visual C++ têm de chamar VariantInit().

Nota: Devido a um erro no Microsoft Excel 7.0 e no Microsoft Access 7.0 (e possivelmente em outros produtos de 32 bits da Microsoft que utilizam o Visual Basic for Applications), dados de cadeia em variáveis Variant recebidos do QMF para Windows podem não ser convertidos de Unicode (usado por OLE) em ANSI (usado por VBA). Nesta situação, só é apresentado o primeiro carácter da cadeia. Para resolver este

problema, defina a variável igual a uma cadeia vazia antes de chamar a função do QMF para Windows que utiliza a variável.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()` para obter informações de erro adicionais.

Tópicos Relacionados

`SetOption()`

GetOptionEx()

`GetOptionEx` curto(*Option* curto)

Descrição

Obtém o valor da opção especificada no QMF para Windows. Quando o valor de opção for devolvido, tem de se consultar a propriedade *Option* acerca dos dados.

Parâmetros

Nome	Descrição
Option	Os valores de opção são os mesmos que os da chamada <code>GetOption()</code> .

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()` para obter informações de erro adicionais.

Tópicos Relacionados

`GetOption()`

`SetOption()`

GetProcText()

`GetProcText` BSTR(*ProcID* longo)

Descrição

Esta função devolve o texto que é executado para o procedimento especificado, depois da substituição de variáveis. Deverá utilizar `SetProcVariable()` para definir o valor de quaisquer variáveis usadas no procedimento antes de chamar esta função.

Parâmetros

Nome	Descrição
------	-----------

ProcID	O ID do procedimento tal como é devolvido por InitializeProc().
--------	---

Valor de Retorno

Se tiver êxito, será devolvida uma cadeia contendo o texto do procedimento. Caso contrário, é devolvido NULL. Se o valor de retorno for NULL, pode-se chamar GetLastErrorString() ou GetLastErrorType() para obter informações de erro adicionais.

GetProcAddress()

GetProcAddress curto(*ProcID* longo, *VARIANT* Variables*)

Descrição

Obtém o valor da opção especificada no QMF para Windows.

Parâmetros

Nome	Descrição
ProcID	O ID do procedimento tal como é devolvido por InitializeProc().
Value	Um apontador para uma VARIANT na qual o resultado está armazenado. O resultado será uma matriz de cadeias (tipo de variante VT_ARRAY VT_BSTR) com cada cadeia contendo o nome de uma variável. Se não houver variáveis no procedimento, o resultado é vazio (tipo de variante VT_EMPTY). Tem de iniciar devidamente a VARIANT antes de chamar esta função. O Visual Basic faz isto automaticamente. Os programadores de Visual C++ têm de chamar VariantInit().

Nota: Devido a um erro no Microsoft Excel 7.0 e no Microsoft Access 7.0 (e possivelmente em outros produtos de 32 bits da Microsoft que utilizam o Visual Basic for Applications), dados de cadeia em variáveis Variant recebidos do QMF para Windows podem não ser convertidos de Unicode (usado por OLE) em ANSI (usado por VBA). Nesta situação, só é apresentado o primeiro carácter da cadeia. Para resolver este problema, defina a variável igual a uma cadeia vazia antes de chamar a função do QMF para Windows que utiliza a variável.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se não houver variáveis no procedimento, o valor de retorno é RS_NO_ERROR_NO_DATA (-1). Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType() para obter informações de erro adicionais.

GetQMFObjectInfo()

GetQMFObjectInfo curto(*OwnerAndName* BSTR, *Type* curto, *Time* curto, VARIANT* *Value*)

Descrição

Esta função devolve informações acerca de um objecto do QMF (um formulário ou uma consulta). As informações devolvidas são especificadas pelos parâmetros *Type* e *Time*.

Parâmetros

Nome	Descrição
OwnerAndName	Uma cadeia contendo o proprietário e o nome, separados por um ponto, do objecto para o qual pretende obter informações. Por exemplo, João.Consulta2
Value	Um apontador para uma VARIANT na qual o resultado está armazenado. No caso de RSI_TIMEUSED, RSI_TIMESRUN, RSI_TIMESCANCELLED e RSI_LEVEL, o resultado é um curto (tipo de variante VT_I2). No caso de RSI_RESTRICTED o resultado é um Booleano (tipo de variante VT_BOOL). Em todos os outros casos, o resultado é uma cadeia (tipo de variante VT_BSTR). Tem de iniciar devidamente a VARIANT antes de chamar esta função. O Visual Basic faz isto automaticamente. Os programadores de Visual C++ têm de chamar VariantInit().

Nota: Devido a um erro no Microsoft Excel 7.0 e no Microsoft Access 7.0 (e possivelmente em outros produtos de 32 bits da Microsoft que utilizam o Visual Basic for Applications), dados de cadeia em variáveis Variant recebidos do QMF para Windows podem não ser convertidos de Unicode (usado por OLE) em ANSI (usado por VBA). Nesta situação, só é apresentado o primeiro carácter da cadeia. Para resolver este problema, defina a variável igual a uma cadeia vazia antes de chamar a função do QMF para Windows que utiliza a variável.

<i>Type</i>	Especifica o tipo de informações a obter.
Valor	Significado
0 (RSI_COMMENT)	Comentário
1 (RSI_LEVEL)	Nível
2 (RSI_TYPE)	Tipo
3 (RSI_SUBTYPE)	Subtipo

4 (RSI_RESTRICTED)	Restrito
5 (RSI_MODEL)	Modelo
6 (RSI_TIMESUSED)	Número de vezes usado.
7 (RSI_TIMESRUN)	Número de vezes executado.
8 (RSI_TIMESCANCELLED)	Número de vezes cancelado.
9 (RSI_DATE)	Data da primeira utilização, da última utilização ou modificação.
10 (RSI_TIME)	Hora da primeira utilização, da última utilização ou modificação.
11 (RSI_USERID)	Primeira utilização, última utilização ou modificação do ID de utilizador.
12 (RSI_SQLID)	Primeira utilização, última utilização ou modificação do ID de SQL.
13 (RSI_ENVIRONMENT)	Primeira utilização, última utilização ou modificação do ambiente.
14 (RSI_MODE)	Primeira utilização, última utilização ou modificação do modo.
15 (RSI_COMMAND)	Primeira utilização, última utilização ou modificação do comando.
<i>Time</i>	Especifica a primeira utilização, última utilização ou modificação do comando.
Valor	Significado
0 (RST_FIRSTUSED)	Primeira utilização.
1 (RST_LASTUSED)	Última utilização.
2 (RST_LASTMODIFIED)	Última modificação.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

GetQMFObjectInfoEx()

`GetQMFObjectInfoEx` curto(*OwnerAndName* BSTR, *Type* curto, *Time* curto)

Descrição

Esta função devolve informações acerca de um objecto do QMF. As informações devolvidas são especificadas pelos parâmetros *Type* e *Time*. Depois de chamar esta função, a propriedade *QMFObjectInfo* pode ser interrogada acerca do valor devolvido.

Parâmetros

Nome	Descrição
OwnerAndName	Uma cadeia contendo o proprietário e o nome, separados por um ponto, do objecto para o qual pretende obter informações. Por exemplo, João.Consulta2
Type	Especifica o tipo de informações a obter.

Valor	Significado
0 (RSI_COMMENT)	Comentário
1 (RSI_LEVEL)	Nível
2 (RSI_TYPE)	Tipo
3 (RSI_SUBTYPE)	Subtipo
4 (RSI_RESTRICTED)	Restrito
5 (RSI_MODEL)	Modelo
6 (RSI_TIMESUSED)	Número de vezes usado.
7 (RSI_TIMESRUN)	Número de vezes executado.
8 (RSI_TIMESCANCELLED)	Número de vezes cancelado.
9 (RSI_DATE)	Data da primeira utilização, da última utilização ou modificação.
10 (RSI_TIME)	Hora da primeira utilização, da última utilização ou modificação.
11 (RSI_USERID)	Primeira utilização, última utilização ou modificação do ID de utilizador.
12 (RSI_SQLID)	Primeira utilização, última utilização ou modificação do ID de SQL.
13 (RSI_ENVIRONMENT)	Primeira utilização, última utilização ou modificação do ambiente.
14 (RSI_MODE)	Primeira utilização, última utilização ou modificação do modo.
15 (RSI_COMMAND)	Primeira utilização, última utilização ou modificação do comando.

<i>Time</i>	Especifica a primeira utilização, última utilização ou modificação do comando.
Valor	Significado
0 (RST_FIRSTUSED)	Primeira utilização.
1 (RST_LASTUSED)	Última utilização.
2 (RST_LASTMODIFIED)	Última modificação.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

GetQMFOBJECTList()

`GetQMFOBJECTList` curto(*Owner* BSTR, *Name* BSTR, *Type* curto, VARIANT* *List*)

Descrição

Esta função devolve uma matriz dos nomes de todos os objectos do QMF correspondentes aos padrões especificados nos parâmetros *Owner* e *Name*.

Parâmetros

Nome	Descrição
Owner	Uma cadeia contendo o proprietário dos objectos que pretende incluir na lista devolvida.
Name	Uma cadeia contendo o nome dos objectos que pretende incluir na lista devolvida.
List	Um apontador para uma VARIANT na qual o resultado está armazenado. O resultado será uma matriz de cadeias (tipo de variante VT_ARRAY VT_BSTR) cada qual com o formato Proprietário.Nome. Se não forem encontradas consultas do QMF para Windows correspondentes, o resultado é vazio (tipo de variante VT_EMPTY). Tem de iniciar devidamente a VARIANT antes de chamar esta função. O Visual Basic faz isto automaticamente. Os programadores de Visual C++ têm de chamar <code>VariantInit()</code> .

Nota: Devido a um erro no Microsoft Excel 7.0 e no Microsoft Access 7.0 (e possivelmente em outros produtos de 32 bits da Microsoft que utilizam o Visual Basic for Applications), dados de cadeia em variáveis Variant recebidos do QMF para Windows podem não ser convertidos de Unicode (usado por OLE) em ANSI (usado por VBA). Nesta situação,

só é apresentado o primeiro carácter da cadeia. Para resolver este problema, defina a variável igual a uma cadeia vazia antes de chamar a função do QMF para Windows que utiliza a variável.

<i>Type</i>	Especifica os tipos de objectos do QMF que pretende incluir na lista. Estes valores podem ser adicionados juntos para especificar tipos de objectos múltiplos.
Valor	Significado
2048 (RSQ_MASK_QUERIES)	Incluir consultas do QMF na lista.
1024 (RSQ_MASK_FORMS)	Incluir formulários do QMF na lista.
512 (RSQ_MASK_PROCS)	Incluir procedimentos do QMF na lista.
256 (RSQ_MASK_TABLES)	Incluir tabelas na lista.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se não forem encontrados objectos do QMF correspondentes, o valor de retorno é zero. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

GetQMFOBJECTListEx()

`GetQMFOBJECTListEx` curto(*Owner* BSTR, *Name* BSTR, *Index* curto)

Descrição

Esta função devolve o nome do objecto do QMF correspondente aos padrões especificados nos parâmetros *Owner* e *Name* referenciados pelo parâmetro *Index*. Depois de chamar esta função, a propriedade *Value* pode ser interrogada acerca do valor devolvido.

Parâmetros

Nome	Descrição
Owner	Uma cadeia contendo o proprietário dos objectos que pretende incluir na lista devolvida.
Name	Uma cadeia contendo o nome dos objectos que pretende incluir na lista devolvida.
Index	O índice da lista de objectos do QMF que correspondem ao padrão.
<i>Type</i>	Especifica os tipos de objectos do QMF que pretende incluir na lista. Estes valores podem ser adicionados juntos para especificar tipos de objectos múltiplos.

Valor	Significado
2048 (RSQ_MASK_QUERIES)	Incluir consultas do QMF na lista.
1024 (RSQ_MASK_FORMS)	Incluir formulários do QMF na lista.
512 (RSQ_MASK_PROCS)	Incluir procedimentos do QMF na lista.
256 (RSQ_MASK_TABLES)	Incluir tabelas na lista.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se não forem encontrados objectos do QMF correspondentes, o valor de retorno será `RS_ERROR_OUTOFRANGE`. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

GetQMFProcText()

GetQMFProcText BSTR(*OwnerAndName* BSTR)

Descrição

Esta função devolve o texto que é executado para o procedimento especificado, depois da substituição de variáveis. Deverá utilizar `SetProcVariable()` para definir o valor de quaisquer variáveis usadas no procedimento antes de chamar esta função.

Parâmetros

Nome	Descrição
OwnerAndName	Uma cadeia contendo o proprietário e o nome, separados por um ponto, do objecto que pretende eliminar. Por exemplo, João.Proc2

Valor de Retorno

Uma cadeia contendo o texto para o procedimento que foi obtido, ou NULL se não tiver sido possível obter o procedimento. Se o valor de retorno for NULL, pode-se chamar `GetLastErrorString()`, `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()` ou `GetLastSQLState()` para obter informações de erro adicionais.

GetQMFQueryText()

GetQMFQueryText BSTR(*OwnerAndName* BSTR)

Descrição

Esta função obtém o texto SQL armazenado na consulta especificada.

Parâmetros

Nome	Descrição
OwnerAndName	Uma cadeia contendo o proprietário e o nome, separados por um ponto, do objecto que pretende eliminar. Por exemplo, João.Consulta2

Valor de Retorno

Uma cadeia contendo o texto para a consulta que foi obtida, ou NULL se não tiver sido possível obter a consulta. Se o valor de retorno for NULL, pode-se chamar `GetLastErrorString()`, `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()` ou `GetLastSQLState()` para obter informações de erro adicionais.

GetQueryText()

GetQueryText BSTR(*QueryID* longo)

Descrição

Esta função devolve o texto SQL que é executado para a consulta especificada, depois da substituição de variáveis. Deverá utilizar `SetVariable()` para definir o valor de quaisquer variáveis usadas na consulta antes de chamar esta função.

Parâmetros

Nome	Descrição
QueryID	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .

Nota: O texto da consulta não está disponível para instruções SQL estáticas. No caso de IDs de consulta devolvidos por `InitializeStaticQuery()`, `GetQueryText()` devolve uma cadeia vazia.

Valor de Retorno

Se tiver êxito, será devolvida uma cadeia contendo o texto SQL. Caso contrário, é devolvido NULL. Se o valor de retorno for NULL, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()` para obter informações de erro adicionais.

GetQueryVerb()

GetQueryVerb BSTR(*QueryID* longo)

Descrição

Esta função devolve uma cadeia contendo o verbo SQL que utilizou na consulta.

Parâmetros

Nome	Descrição
QueryID	O ID da consulta tal como é devolvido por InitializeQuery().

Nota: O verbo da consulta não está disponível para instruções SQL estáticas. No caso de IDs de consulta devolvidos por InitializeStaticQuery(), GetQueryVerb() devolve uma cadeia vazia.

Valor de Retorno

Se tiver êxito, será devolvida uma cadeia contendo o verbo SQL. Caso contrário, é devolvido NULL. Se o valor de retorno for NULL, pode-se chamar GetLastErrorString() ou GetLastErrorType() para obter informações de erro adicionais.

GetResourceLimit()

GetResourceLimit curto(*Resource* curto, *Value* longo*)

Descrição

Esta função obtém o limite de recurso solicitado. Tem de chamar InitializeServer() antes de chamar esta função, dado que os limites de recurso são processados por servidor.

Parâmetros

Nome	Descrição
Resource	Os valores de recurso incluem:

Valor	Significado
0 (RSR_IDLE_CONNECTION_TIMEOUT)	Tempo de espera de ligação inactiva (em segundos).
1 (RSR_IDLE_QUERY_TIMEOUT_CANCEL)	Tempo de espera de consulta inactiva (em segundos).
2 (RSR_IDLE_QUERY_TIMEOUT_WARNING)	Tempo de espera de consulta inactiva (em segundos). Este é o limite de aviso; não está em vigor para a API do QMF para Windows.
3 (RSR_SERVER_RESPONSE_TIMEOUT_CANCEL)	Tempo de espera de servidor (em segundos)

4 (RSR_SERVER_RESPONSE_TIMEOUT_WARNING)	Tempo de espera de servidor (em segundos) Este é o limite de aviso; não está em vigor para a API do QMF para Windows.
5 (RSR_MAX_ROWS_TO_FETCH_CANCEL)	Número máximo de linhas a buscar.
6 (RSR_MAX_ROWS_TO_FETCH_WARNING)	Número máximo de linhas a buscar. Este é o limite de aviso; não está em vigor para a API do QMF para Windows.
7 (RSR_MAX_BYTES_TO_FETCH_CANCEL)	Número máximo de bytes a buscar.
8 (RSR_MAX_BYTES_TO_FETCH_WARNING)	Número máximo de bytes a buscar. Este é o limite de aviso; não está em vigor para a API do QMF para Windows.
9 (RSR_MAX_CONNECTIONS)	Número máximo de ligações permitidas ao servidor de base de dados.
10 (RSR_ALLOW_SERVER_ACCESS_UI)	É permitido o acesso ao servidor de base de dados por parte da interface do QMF para Windows?
11 (RSR_ALLOW_SERVER_ACCESS_API)	É permitido o acesso ao servidor de base de dados por parte da API do QMF para Windows?
12 (RSR_FETCH_ALL_ROWS)	Buscar automaticamente todas as linhas?
13 (RSR_CONFIRM_UPDATES)	Confirmar actualizações ao servidor de base de dados? Esta opção não tem efeito na API do QMF para Windows; as actualizações à base de dados não são confirmadas para a API do QMF para Windows.
14 (RSR_SUMMARY_TRACKING)	Está activada a visualização imediata de objectos de resumo?

15 (RSR_DETAILED_TRACKING)	Está activada a visualização imediata de objectos detalhados?
16 (RSR_SQL_TRACKING)	Está activada a visualização imediata de texto SQL?
17 (RSR_ADHOC_TRACKING)	Está activada a visualização imediata de objectos ad hoc?
18 (RSR_ALLOW_ACQUIRE)	É permitido o verbo SQL ACQUIRE?
19 (RSR_ALLOW_ALTER)	É permitido o verbo SQL ALTER?
20 (RSR_ALLOW_COMMENT)	É permitido o verbo SQL COMMENT?
21 (RSR_ALLOW_CREATE)	É permitido o verbo SQL CREATE?
22 (RSR_ALLOW_DELETE)	É permitido o verbo SQL DELETE?
23 (RSR_ALLOW_DROP)	É permitido o verbo SQL DROP?
24 (RSR_ALLOW_EXPLAIN)	É permitido o verbo SQL EXPLAIN?
25 (RSR_ALLOW_GRANT)	É permitido o verbo SQL GRANT?
26 (RSR_ALLOW_INSERT)	É permitido o verbo SQL INSERT?
27 (RSR_ALLOW_LABEL)	É permitido o verbo SQL LABEL?
28 (RSR_ALLOW_LOCK)	É permitido o verbo SQL LOCK?
29 (RSR_ALLOW_REVOKE)	É permitido o verbo SQL REVOKE?
30 (RSR_ALLOW_SELECT)	É permitido o verbo SQL SELECT?
31 (RSR_ALLOW_SET)	É permitido o verbo SQL SET?
32 (RSR_ALLOW_SIGNAL)	É permitido o verbo SQL SIGNAL?
33 (RSR_ALLOW_UPDATE)	É permitido o verbo SQL UPDATE?

34 (RSR_ALLOW_CALL)	É permitido o verbo SQL CALL?
35 (RSR_ALLOW_SAVE_DATA)	É permitido o comando Guardar Dados?
36 (RSR_SAVE_DATA_TABLE_SPACE_NAME)	O nome de coleção predefinido para associar pacotes?
37 (RSR_SAVE_DATA_TABLE_SPACE_NAME_OVERRIDE)	O nome de espaço de tabela predefinido para o comando Guardar Dados pode ser sobreposto pelo utilizador?
38 (RSR_ALLOW_BIND_PACKAGE)	Permitir associação de pacotes?
39 (RSR_DEF_COLLECTION)	O nome de coleção predefinido para associar pacotes.
40 (RSR_DEF_COLLECTION_OVERRIDE)	O nome de coleção predefinido para associar pacotes pode ser sobreposto pelo utilizador?
41 (RSR_DEF_ISOLATION_LEVEL)	O nível de isolamento predefinido para associar pacotes.
42 (RSR_DEF_ISOLATION_LEVEL_OVERRIDE)	O nível de isolamento predefinido para associar pacotes pode ser sobreposto pelo utilizador?
43 (RSR_ALLOW_TABLE_EDIT)	Permitir utilização do editor de tabelas?
44 (RSR_ALLOW_EXPORT)	Permitir exportação de dados?
45 (RSR_ALLOW_SAVED_QUERIES_ONLY)	Está o utilizador autorizado a executar somente consultas guardadas?
46 (RSR_ALLOW_DROP_PACKAGE)	Permitir abandono de pacotes?
47 (RSR_QUERY_ISOLATION_LEVEL)	O nível de isolamento a utilizar ao executar consultas.

48 (RSR_ACCOUNT_STRING)	A cadeia especificando informações de conta a transmitir ao efectuar ligação ao servidor de base de dados.
49 (RSR_ACCOUNT_OVERRIDE)	A cadeia que especifica informações de conta a transmitir ao efectuar ligação ao servidor de base de dados pode ser sobreposta pelo utilizador?
Value	Um apontador para um longo no qual o resultado está armazenado. O resultado é o valor do limite de recurso solicitado. No caso de valores Booleanos, o resultado é diferente de zero para verdadeiro e zero para falso. No caso de RSR_SAVE_DATA_TABLE_SPACE_NAME, RSR_DEF_COLLECTION e RSR_ACCOUNT_STRING, é devolvido -1 e a propriedade <i>ResourceLimit</i> pode ser interrogada acerca do valor da cadeia devolvida.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()` para obter informações de erro adicionais.

GetResourceLimitEx()

`GetResourceLimitEx` curto(*Resource* curto)

Descrição

Esta função obtém o limite de recurso solicitado. Tem de chamar `InitializeServer()` antes de chamar esta função, dado que os limites de recurso são processados por servidor. Após uma chamada desta função, consulte a propriedade *ResourceLimit* acerca do resultado.

Parâmetros

Nome	Descrição
Resource	Os valores de recurso são os mesmos que os da chamada <code>GetResourceLimit()</code> .

Nota: O verbo da consulta não está disponível para instruções SQL estáticas. No caso de IDs de consulta devolvidos por `InitializeStaticQuery()`, `GetQueryVerb()` devolve uma cadeia vazia.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()` para obter informações de erro adicionais.

GetRowCount()

`GetRowCount` longo(*QueryID* longo)

Descrição

Esta função devolve o número de linhas actualmente na memória tampão interna do QMF para Windows. Tal pode verificar-se superior ao número de linhas obtidas com `FetchNextRow()` ou `FetchNextRows()`, dado que o QMF para Windows coloca em memória tampão dados recebidos da base de dados.

Esta função devolve o número de linhas já obtidas da base de dados. Caso prefira obter o número total de linhas no conjunto de resultados, poderá:

- Chamar `CompleteQuery()` e buscar todas as linhas usando `FetchNextRow()` ou `FetchNextRows()`.
- Especificar `FetchAllRows = TRUE` ao chamar `Open()`.

Parâmetros

Nome	Descrição
<code>QueryID</code>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .

Valor de Retorno

O número de linhas se tiver êxito (0 se não tiverem sido obtidas linhas) ou -1 se não tiver êxito. Se for 1, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()` para obter informações de erro adicionais.

GetServerList()

`GetServerList` curto(*VARIANT* List*)

Descrição

Esta função devolve uma matriz contendo os nomes dos servidores de base de dados definidos no Ficheiro de Definição de Servidores (SDF) do QMF para Windows. Tem de definir um servidor de base de dados no ficheiro SDF se lhe quiser ter aceder através da API do QMF para Windows.

Parâmetros

Nome	Descrição
------	-----------

List	Um apontador para uma VARIANT na qual o resultado está armazenado. O resultado será uma matriz de cadeias (tipo de variante VT_ARRAY VT_BSTR) com cada cadeia contendo o nome de um servidor de base de dados. Se não tiver definido servidores de base de dados nenhuns, o resultado será vazio (tipo de variante VT_EMPTY). Tem de iniciar devidamente a VARIANT antes de chamar esta função. O Visual Basic faz isto automaticamente. Os programadores de Visual C++ têm de chamar VariantInit().
------	--

Nota: Devido a um erro no Microsoft Excel 7.0 e no Microsoft Access 7.0 (e possivelmente em outros produtos de 32 bits da Microsoft que utilizam o Visual Basic for Applications), dados de cadeia em variáveis Variant recebidos do QMF para Windows podem não ser convertidos de Unicode (usado por OLE) em ANSI (usado por VBA). Nesta situação, só é apresentado o primeiro carácter da cadeia. Para resolver este problema, defina a variável igual a uma cadeia vazia antes de chamar a função do QMF para Windows que utiliza a variável.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se não tiver definido servidores de base de dados nenhuns, o valor de retorno será zero. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType() para obter informações de erro adicionais.

GetServerListEx()

GetServerListEx curto(*Index* curto)

Descrição

Esta função obtém o nome do servidor referenciado pelo parâmetro *Index*. Depois de chamar esta função, a propriedade *Value* pode ser interrogada acerca do valor devolvido.

Parâmetros

Nome	Descrição
Index	Um índice na lista de servidores.

Valor de Retorno

Zero se tiver êxito, RS_OUTOFRANGE quando o índice é superior ao número de servidores disponíveis, diferente de zero se não tiver êxito. Se não tiver definido servidores de base de dados nenhuns, o valor de retorno será RS_OUTOFRANGE. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType() para obter informações de erro adicionais.

GetStoredProcedureResultSets()

GetStoredProcedureResultSets curto(*QueryID* longo, VARIANT* *ResultSets*)

Descrição

Esta função obtém os IDs de consulta para os conjuntos de resultados devolvidos pelo procedimento armazenado executado com o *QueryID* original. Cada ID de consulta devolvido pode ser usado com FetchNextRow() ou FetchNextRows() para obter as linhas do conjunto de resultados e Close() quando se atinge o final de cada conjunto de resultados.

Parâmetros

Nome	Descrição
QueryID	O ID da consulta original tal como é devolvido por InitializeQuery().
ResultSets	Um apontador para uma VARIANT na qual estão armazenados os IDs de consulta para os conjuntos de resultados. O resultado será uma matriz de números inteiros longos (tipo de variante VT_ARRAY VT_I4), com cada qual sendo o ID de consulta relativo aos conjuntos de resultados correspondentes. Se o procedimento armazenado não tiver devolvido conjuntos de resultados nenhuns, o resultado será vazio (tipo de variante VT_EMPTY). Tem de iniciar devidamente a VARIANT antes de chamar esta função. O Visual Basic faz isto automaticamente. Os programadores de Visual C++ têm de chamar VariantInit().

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType(), GetLastErrorSQLCode(), GetLastErrorSQLError(), ou GetLastErrorSQLState() para obter informações de erro adicionais.

GetVariables()

GetVariables curto(*QueryID* longo, VARIANT* *Variables*)

Descrição

Esta função devolve uma matriz dos nomes das variáveis no texto SQL da consulta. Tem de se atribuir valores a estas variáveis, chamando SetVariable() antes de executar a consulta com Open() ou Execute().

Parâmetros

Nome	Descrição
QueryID	O ID da consulta tal como é devolvido por InitializeQuery().

Variables	Um apontador para uma VARIANT na qual o resultado está armazenado. O resultado será uma matriz de cadeias (tipo de variante VT_ARRAY VT_BSTR) com cada cadeia contendo o nome de uma variável. Se não houver variáveis na instrução SQL o resultado é vazio (tipo de variante VT_EMPTY). Tem de iniciar devidamente a VARIANT antes de chamar esta função. O Visual Basic faz isto automaticamente. Os programadores de Visual C++ têm de chamar VariantInit().
-----------	---

Nota: Devido a um erro no Microsoft Excel 7.0 e no Microsoft Access 7.0 (e possivelmente em outros produtos de 32 bits da Microsoft que utilizam o Visual Basic for Applications), dados de cadeia em variáveis Variant recebidos do QMF para Windows podem não ser convertidos de Unicode (usado por OLE) em ANSI (usado por VBA). Nesta situação, só é apresentado o primeiro carácter da cadeia. Para resolver este problema, defina a variável igual a uma cadeia vazia antes de chamar a função do QMF para Windows que utiliza a variável.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se não houver variáveis na instrução SQL, o valor de retorno é RS_ERROR_NO_DATA (-1). Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType() para obter informações de erro adicionais.

GetVariablesEx()

GetVariablesEx curto(*QueryID* longo, *Index* curto)

Descrição

Esta função devolve o nome da variável no texto SQL da consulta referenciada pelo parâmetro *Index*. Depois de chamar esta função, a propriedade *Value* pode ser interrogada acerca do valor devolvido. Tem de se atribuir valores a esta variável (e a todas as outras no texto SQL) chamando SetVariable() antes de executar a consulta com Open() ou Execute().

Parâmetros

Nome	Descrição
QueryID	O ID da consulta tal como é devolvido por InitializeQuery().
Variables	Um índice na lista interna de variáveis. Consulte com a propriedade <i>Value</i> para saber qual a cadeia que corresponde ao índice transmitido. Se não existirem variáveis na instrução SQL, a função devolve RS_ERROR_NO_DATA.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se não houver variáveis na instrução SQL, o valor de retorno é RS_ERROR_NO_DATA (-1). Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType() para obter informações de erro adicionais.

InitializeProc()

InitializeProc longo(*SourceType* curto, *Source* BSTR)

Descrição

Esta função define o texto que pretende utilizar num procedimento. Pode-se transmitir o texto como um parâmetro a esta função, lê-lo a partir de um ficheiro de texto ou obtê-lo de um procedimento existente.

Parâmetros

Nome	Descrição
SourceType	Especifica a origem do texto de procedimento.
Valor	Significado
0 (RSS_STRING)	O texto está contido no parâmetro <i>Source</i> .
2 (RSS_FILE)	O texto está contido no ficheiro de texto cujo nome é especificado pelo parâmetro <i>Source</i> .
3 (RSS_QMFPROC)	O texto está contido no procedimento cujos proprietário e nome são especificados pelo parâmetro <i>Source</i> .
<i>Source</i>	Uma cadeia contendo o texto, o proprietário e o nome (Nome.Proprietário) do procedimento, ou o nome de um ficheiro contendo o texto de procedimento.

Valor de Retorno

Se tiver êxito, o ID do procedimento (ProcID). Caso contrário, -1. Tem de se usar este valor em todas as chamadas de interface que exijam o parâmetro *ProcID*.

InitializeQuery()

InitializeQuery longo(*SourceType* curto, *Source* BSTR)

Descrição

Esta função define o texto que pretende utilizar numa consulta. Pode-se transmitir o texto SQL como um parâmetro a esta função, lê-lo a partir de um ficheiro de texto ou obtê-lo de uma consulta existente. Chame Close() quando concluir esta consulta.

Parâmetros

Nome	Descrição
SourceType	Especifica a origem do texto de instrução SQL.
Valor	Significado
0 (RSS_STRING)	O texto SQL está contido no parâmetro <i>Source</i> .
1 (RSS_QMFQUERY)	O texto SQL está contido na consulta cujos proprietário e nome são especificados pelo parâmetro <i>Source</i> .
2 (RSS_FILE)	O texto SQL está contido no ficheiro de texto cujo nome é especificado pelo parâmetro <i>Source</i> .

Valor de Retorno

Se tiver êxito, o ID da consulta. Caso contrário, -1. Tem de se usar este valor em todas as chamadas de interface que exijam o parâmetro *Query*.

InitializeServer()

InitializeServer curto(*ServerName* BSTR, *UserID* BSTR, *Password* BSTR, *ForceDialog* BOOL, [VARIANT *Account*], [VARIANT *SuppressDialog*])

Descrição

Esta função inicia uma ligação a um servidor de base de dados. Tem de se chamar esta função antes de chamar qualquer outra função na API do QMF para Windows. Pode-se chamar esta função várias vezes. Todavia, se chamar esta função e não concluir chamando Commit() ou Rollback(), resultará numa remoção de alterações implícita.

Parâmetros

Nome	Descrição
ServerName	Uma cadeia contendo o nome do servidor de base de dados que pretende utilizar. Este nome tem de corresponder a um dos nomes definidos no Ficheiro de Definição de Servidores do QMF para Windows. Chame GetServerList() para obter uma lista de servidores válidos.

UserID	Uma cadeia contendo o ID de Utilizador que pretende usar. Se UserID for NULL ou uma cadeia vazia, o QMF para Windows tentará utilizar o ID de Utilizador da consulta mais recente, se estiver disponível. Caso contrário, o QMF para Windows apresentará a caixa de diálogo Informações de Utilizador para obter um ID de Utilizador e uma palavra-passe.
Password	Uma cadeia contendo a palavra-passe para o ID de utilizador especificado. Se a Password for NULL ou uma cadeia vazia, o QMF para Windows tentará utilizar uma palavra-passe memorizada, se estiver disponível (exige Windows para Workgroups). Se não estiver disponível nenhuma palavra-passe, o QMF para Windows apresentará a caixa de diálogo Informações de Utilizador para obter uma palavra-passe.
ForceDialog	Um valor diferente de zero indica que o QMF para Windows apresenta a caixa de diálogo Informações de Utilizador quer tenham sido especificados, ou não, um ID de Utilizador e uma Palavra-passe. Permite-se assim ao utilizador alterar as informações antes destas serem utilizadas. Zero indica que o QMF para Windows deveria apresentar a caixa de diálogo Informações de Utilizador somente quando for necessário.
Account	A título opcional, uma cadeia especificando informações de conta a transmitir ao servidor ao efectuar ligação. O servidor pode utilizar esta informação num sistema de contabilização do trabalho.
SuppressDialog	Um valor diferente de zero indica que o QMF para Windows não apresenta a caixa de diálogo Informações de Utilizador, mesmo que não tenham sido especificados um ID de Utilizador e uma Palavra-passe. Esta opção é útil ao executar num ambiente sem utilizadores para responder à caixa de diálogo Informações de Utilizador, por exemplo, num servidor da Web.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType(), GetLastErrorSQLCode(), GetLastErrorSQLError(), ou GetLastErrorSQLState() para obter informações de erro adicionais.

Tópicos Relacionados

SetParent()

InitializeStaticQuery()

InitializeStaticQuery longo(*CollectionName* BSTR, *PackageName* BSTR, *ConsistencyToken* BSTR, *SectionNumber* curto)

Descrição

Esta função especifica a secção de um pacote que pretende executar como uma consulta estática.

Parâmetros

Nome	Descrição
CollectionName	O nome de uma colecção anteriormente associada.
PackageName	O nome de um pacote anteriormente associado.
ConsistencyToken	O símbolo usado pelo pacote e pela colecção acima referidos.
SectionNumber	O número de secção da instrução dentro da colecção e do pacote que pretende executar.

Valor de Retorno

Se tiver êxito, o ID da consulta. Caso contrário, -1. Tem de se usar este valor em todas as chamadas de interface que exijam o parâmetro *QueryID*.

IsStatic()

IsStatic BOOL(*QueryID* longo)

Descrição

Esta função determina se o ID de consulta especificado se refere a uma consulta estática ou dinâmica.

Parâmetros

Nome	Descrição
QueryID	O ID da consulta tal como é devolvido por InitializeQuery() ou InitializeStaticQuery().

Valor de Retorno

Devolve um valor diferente de zero se tiver êxito e se *QueryID* se referir a uma consulta estática, caso contrário, zero.

Open()

Open curto(*QueryID* longo, *RowLimit* longo, *FetchAllRows* BOOL)

Descrição

Utilize esta função para executar uma consulta que use o verbo SELECT, abrindo um cursor na base de dados para a consulta. Use FetchNextRow() ou FetchNextRows() para obter os dados para a consulta e chame Close() quando terminar. Se o QMF para Windows estiver configurado para buscar automaticamente todas as linhas (consulte RSR_FETCHALLROWS na descrição para GetResourceLimit()) ou se o parâmetro FetchAllRows for

diferente de zero, o QMF para Windows busca todas as linhas do conjunto de resultados para a sua memória tampão interna antes de regressar desta chamada.

Nota: O nome desta função entra em conflito com a palavra-chave `Open` do Microsoft Access 2.0. Se utilizar MS Access 2.0, coloque parêntesis rectos [] em redor do nome da função.

Nota: Use esta função somente em instruções que contenham o verbo SQL `SELECT`. No caso de instruções contendo qualquer outro verbo, por exemplo `SET`, chame `Execute()`. Para saber qual o verbo usado por uma consulta, chame `GetQueryVerb()`.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .
<i>RowLimit</i>	Um número indicando o número máximo de linhas a obter da base de dados. Zero indica que não vigora limite algum, excepto o limite de linhas estabelecido pelo programa Administrador do QMF para Windows.
<i>FetchAllRows</i>	Um valor Booleano que indica se são ou não buscadas automaticamente todas as linhas no conjunto de resultados para a memória tampão interna do QMF para Windows. Se for diferente de zero, todas as linhas são automaticamente buscadas, fechando o cursor e libertando a base de dados para utilização por parte de outros elementos. Esta acção equivale a chamar <code>CompleteQuery()</code> .

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

Prepare()

Prepare curto(*QueryID* longo)

Descrição

Esta função prepara a consulta especificada por *QueryID*. A instrução é examinada pelo servidor de base de dados, verificando a existência de objectos, autorizações necessárias, etc. Se a consulta for uma instrução `SELECT`, estarão disponíveis informações acerca das colunas devolvidas pela instrução depois de concluir `Prepare()`.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por InitializeQuery().

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType(), GetLastSQLCode(), GetLastSQLError(), ou GetLastSQLState() para obter informações de erro adicionais.

Tópicos Relacionados

Execute()

Open()

PrintReport()

PrintReport curto(*QueryID* longo, *SourceType* curto, *Source* BSTR, *OutputFileName* BSTR, *PageLength* curto, *PageWidth* curto, *IncludeDateTime* BOOL, *IncludePageNumbers* BOOL, [VARIANT *Format*], [VARIANT *UseFormPageSetup*])

Descrição

PrintReport() é sinónimo de ExportReport().

ReinitializeServer()

ReinitializeServer curto()

Descrição

Esta função reinicia a ligação a um servidor de base de dados. Regra geral, só será preciso chamar esta função se uma das outras funções da API do QMF para Windows devolver um erro. Chamar esta função resulta numa remoção de alterações implícita, a qual fecha cursores abertos e invalida todos os IDs de consulta pendentes.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType(), GetLastSQLCode(), GetLastSQLError(), ou GetLastSQLState() para obter informações de erro adicionais.

Rollback()

Rollback curto()

Descrição

Esta função cancela quaisquer alterações efectuadas na unidade de trabalho actual, finaliza esta última, fecha cursores abertos e invalida todos os IDs de consulta pendentes.

Nota: O nome desta função entra em conflito com a palavra-chave Rollback do Microsoft Access 2.0. Se utilizar MS Access 2.0, coloque parêntesis rectos [] em redor do nome da função.

Nota: A remoção de alterações só afecta alterações SQL que tenham sido executadas chamando Open() ou Execute(). Não afecta alterações efectuadas por outras funções da API do QMF para Windows, tais como FastSaveData(), SaveData() ou DeleteQMFObject().

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType(), GetLastErrorSQLCode(), GetLastErrorSQLError(), ou GetLastErrorSQLState() para obter informações de erro adicionais.

Tópicos Relacionados

Commit()

RunProc()

RunProc curto(*ProcID* longo)

Descrição

Esta função executa o procedimento especificado, o qual é executado até ao fim ou até que ocorra um erro. Não se pode aceder a nenhum dos resultados do procedimento (por exemplo, dados de uma consulta executada) através desta interface de programação. Contudo, estarão disponíveis ficheiros exportados ou dados guardados pelo procedimento depois da execução.

Parâmetros

Nome	Descrição
<i>ProcID</i>	O ID do procedimento tal como é devolvido por InitializeProc().

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType(), GetLastErrorSQLCode(), GetLastErrorSQLError(), ou GetLastErrorSQLState() para obter informações de erro adicionais.

SaveData()

SaveData curto(*QueryID* longo, *FirstRow* longo, *FirstCol* longo, *LastRow* longo, *LastCol* longo, *Replace* BOOL, *TableName* BSTR, *TableSpaceName* BSTR, *ServerName* BSTR, *UserID* BSTR, *Password* BSTR, *ForceDialog* BOOL, [VARIANT *Account*], [VARIANT *Comment*], [VARIANT *CommitScope*])

Descrição

Esta função copia o intervalo especificado de colunas e linhas para a tabela especificada no espaço de tabela indicado. Tem de chamar `CompleteQuery()` antes de chamar esta função caso não tenha obtido dados de linha para todas as linhas que pretende guardar na tabela. Se tentar guardar linhas que não tenham sido obtidas da base de dados, a salvaguarda falhará. Se a tabela já existir, os novos dados têm de ter o mesmo número e tipos de colunas que a tabela existente.

Esta função opera numa unidade de trabalho separada das outras funções de API e os respectivos resultados são automaticamente consolidados. Chamar `Commit()` ou `Rollback()` não produz efeito em alterações que efectue mediante esta função.

Parâmetros

Nome	Descrição
<i>QueryID</i>	O ID da consulta tal como é devolvido por <code>InitializeQuery()</code> .
<i>FirstRow</i>	A primeira linha que pretende incluir na salvaguarda. O valor da primeira linha num conjunto de resultados é 0.
<i>FirstCol</i>	A primeira coluna que pretende incluir na salvaguarda. O valor da primeira coluna num conjunto de resultados é zero.
<i>LastRow</i>	A última linha que pretende incluir na salvaguarda ou -1 se estiverem incluídas todas as linhas. O valor da última linha num conjunto de resultados é um a menos do que o número total de linhas.
<i>LastCol</i>	A última coluna que pretende incluir na salvaguarda ou -1 se estiverem incluídas todas as colunas. O valor da última coluna num conjunto de resultados é um a menos do que o número total de colunas.
<i>Replace</i>	Um valor diferente de zero indica que os dados especificados substituem dados existentes na tabela. Zero indica que os dados especificados são anexados aos dados existentes na tabela.
<i>TableName</i>	O nome da tabela onde são armazenados os dados. Se a tabela não existir, será criada.
<i>TableSpaceName</i>	O nome do espaço de tabela onde esta existe ou será criada. Se <i>TableSpaceName</i> for NULL ou uma cadeia vazia, será utilizado o espaço de tabela predefinido. Se tiver configurado o QMF para Windows para utilizar sempre o espaço de tabela predefinido (consulte <code>RSR_SDDIFFERENTTS</code>) na descrição para <code>GetResourceLimit()</code> , este parâmetro será ignorado.
<i>ServerName</i>	O nome do servidor de base de dados onde está armazenada a tabela. Se <i>ServerName</i> for NULL ou uma cadeia vazia, o servidor que especificar na chamada de <code>InitializeServer()</code> será utilizado e <i>UserID</i> , <i>Password</i> , <i>ForceDialog</i> e <i>Account</i> serão ignorados.

<i>UserID</i>	Se tiver especificado um servidor diferente em <i>ServerName</i> , <i>UserID</i> será o ID de utilizador para esse servidor. Se não tiver especificado um ID de Utilizador, o QMF para Windows utiliza o último ID de utilizador especificado para este servidor, caso esteja disponível, ou apresenta uma caixa de diálogo se assim não for. Este parâmetro é ignorado se <i>ServerName</i> for NULL ou uma cadeia vazia.
<i>Password</i>	Se tiver especificado um servidor diferente em <i>ServerName</i> , <i>Password</i> será a palavra-passe usada para esse servidor. Se não tiver especificado uma palavra-passe, o QMF para Windows utiliza a última palavra-passe especificada para este servidor, caso esteja disponível, ou apresenta uma caixa de diálogo se assim não for. Este parâmetro é ignorado se <i>ServerName</i> for NULL ou uma cadeia vazia.
<i>ForceDialog</i>	Se tiver especificado um servidor diferente em <i>ServerName</i> , um valor diferente de zero força o QMF para Windows a apresentar uma caixa de diálogo a pedir informações de início de sessão, mesmo que tenham sido especificados ID de utilizador e palavra-passe ou estejam de algum modo disponíveis. Zero indica que o QMF para Windows apresenta esta caixa de diálogo somente se for necessário. Este parâmetro é ignorado se <i>ServerName</i> for NULL ou uma cadeia vazia.
<i>Account</i>	Se especificou um servidor diferente em <i>ServerName</i> , a título opcional, uma cadeia especificando informações de conta a transmitir a esse servidor ao efectuar ligação. O servidor pode utilizar esta informação num sistema de contabilização do trabalho. Este parâmetro é ignorado se <i>ServerName</i> for NULL ou uma cadeia vazia.
<i>Comment</i>	A título opcional, uma cadeia que especifica um comentário para a tabela na qual são guardados os dados.
<i>CommitScope</i>	A título opcional, quantas linhas a inserir numa tabela de uma vez antes de consolidar a unidade de trabalho. Ao especificar zero indicará que se devem inserir todas as linhas antes da consolidação. Ao especificar 10 (por exemplo), indicará que se deve efectuar uma consolidação a cada dez linhas inseridas.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais. Se o conjunto de resultados estiver vazio ou não tiverem sido obtidas linhas da base de dados, será devolvido um valor diferente de zero, a menos que `FirstRow=0` e `LastRow=-1`. Neste caso, será devolvido zero e será criada uma tabela vazia.

SaveQMFProc()

SaveQMFProc curto(*OwnerAndName* BSTR, *Text* BSTR, *Comment* BSTR, *Replace* BOOL, *Share* BOOL)

Descrição

Esta função guarda um procedimento num servidor de base de dados.

Parâmetros

Nome	Descrição
<i>OwnerAndName</i>	Uma cadeia contendo o proprietário e o nome, separados por um ponto, do procedimento que pretende guardar. Por exemplo, João.Proc2
Texto	Uma cadeia contendo o texto que pretende guardar no procedimento.
Comment	Uma cadeia contendo qualquer comentário que pretenda guardar com o procedimento. Se não houver comentário, transmita este parâmetro quer como uma cadeia vazia quer como NULL.
Replace	Um valor diferente de zero substitui um procedimento existente com o mesmo nome. Zero cancela a operação se houver um procedimento com o mesmo nome.
Share	Um valor diferente de zero partilha o procedimento com outros utilizadores. Zero não partilha o procedimento com outros utilizadores.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

SaveQMFQuery()

SaveQMFQuery curto(*OwnerAndName* BSTR, *Text* BSTR, *Comment* BSTR, *Replace* BOOL, *Share* BOOL)

Descrição

Esta função guarda uma consulta num servidor de base de dados.

Parâmetros

Nome	Descrição
------	-----------

<i>OwnerAndName</i>	Uma cadeia contendo o proprietário e o nome, separados por um ponto, da consulta que pretende guardar. Por exemplo, João.Consulta2
Texto	Uma cadeia contendo o texto que pretende guardar na consulta.
Comment	Uma cadeia contendo qualquer comentário que pretenda guardar com a consulta. Se não houver comentário, transmita este parâmetro quer como uma cadeia vazia quer como NULL.
Replace	Um valor diferente de zero substitui uma consulta existente com o mesmo nome. Zero cancela a operação se houver uma consulta com o mesmo nome.
Share	Um valor diferente de zero partilha a consulta com outros utilizadores. Zero não partilha a consulta com outros utilizadores.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

SetBindOption()

`SetBindOption` curto(*CollectionName* BSTR, *PackageName* BSTR, *Option* curto, *Value* curto)

Descrição

Esta função define opções para a colecção e o pacote antes de chamar `EndBind()`.

Parâmetros

Nome	Descrição
<i>CollectionName</i>	O ID de colecção do pacote para o qual pretende definir a opção.
<i>PackageName</i>	O nome do pacote para o qual pretende definir a opção.
<i>Option</i>	Uma das opções enumeradas abaixo.
<i>Value</i>	Um valor diferente de zero substitui uma consulta existente com o mesmo nome. Zero cancela a operação se houver uma consulta com o mesmo nome.
<i>Share</i>	Um dos valores enumerados abaixo para a opção especificada.

Seguem-se os significados e valores relativos às diversas opções:

Opção	Significado	Descrição
DDM_PKGRPLOPT(0x211C)	Sinalizador especificando a substituição, ou não, de um pacote existente com o mesmo ID e nome de coleção.	DDM_PKGRPLALW (0x241F) Sim DDM_PKGRPLNA (0x2420) Não
DDM_STTDECDEL(0x2121)	O delimitador usado como vírgula decimal em instruções SQL no pacote.	DDM_DECDELPRD (0x243C) Ponto DDM_DECDELCMA (0x243D) Vírgula
DDM_STTSTRDEL(0x2120)	O delimitador usado para valores de cadeia em instruções SQL no pacote.	DDM_STRDELAP (0x2426) Apóstrofo DDM_STRDELDO (0x2427) Aspas Duplas
DDM_PKGISOLVL(0x2124)	O nível de isolamento para o pacote.	DDM_ISOLVLALL (0x2443) Todos DDM_ISOLVLCHG (0x2441) Alterar DDM_ISOLVLC (0x2442) Estabilidade do Cursor DDM_ISOLVLNC (0x2445) Sem Consolidação DDM_ISOLVLR (0x2444) Leitura Reproduzível
DDM_PKGATHOPT(0x211E)	Sinalizador especificando se se mantêm, ou não, autorizações existentes no pacote.	DDM_PKGATHKP (0x2425) Manter DDM_PKGATHRVK (0x2424) Revogar
DDM_QRYBLKCTL(0x2132)	O método a usar ao buscar linhas de dados para consultas no pacote.	DDM_FIXROWPRC (0x2418) Uma linha de cada vez DDM_LMTBLKPRC (0x2417) Um bloco de cada vez
DDM_RDBRLOPT(0x2129)	Quando libertar recursos de base de dados adquiridos ao executar o pacote.	DDM_RDBRLOSCMM (0x2438) Consolidar DDM_RDBRLOSCNV (0x2439) Libertação de Conversação
DDM_STTDATEFMT(0x2122)	Formato para valores de data obtidos.	DDM_ISODATEFMT (0x2429) ISO DDM_USADATEFMT (0x242A) US DDM_EURDATEFMT (0x242B) Europeu DDM_JISDATEFMT (0x242C) Norma da Indústria Japonesa

DDM_STTTIMFMT(0x2123)	Formato para valores de hora obtidos.	DDM_ISOTIMFMT (0x242E) ISO DDM_USATIMFMT (0x242F) US DDM_EURTIMFMT (0x2430) Europeu DDM_JISTIMFMT (0x2431) Norma da Indústria Japonesa
-----------------------	---------------------------------------	---

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()` para obter informações de erro adicionais.

SetBindOwner()

`SetBindOwner` curto(*CollectionName* BSTR, *PackageName* BSTR, *OwnerID* BSTR)

Descrição

Esta função permite-lhe especificar um proprietário diferente do seu ID de utilizador para o pacote que está a associar. Tal pode revelar-se necessário se o ID de utilizador não dispuser das autorizações necessárias à associação do pacote mas o proprietário sim.

Parâmetros

Nome	Descrição
<code>CollectionName</code>	O ID de colecção do pacote para o qual pretende especificar o proprietário.
<code>PackageName</code>	O nome do pacote para o qual pretende especificar o proprietário.
<code>Comment</code>	Uma cadeia contendo qualquer comentário que pretenda guardar com a consulta. Se não houver comentário, transmita este parâmetro quer como uma cadeia vazia quer como NULL.
<code>OwnerID</code>	O ID de proprietário pretendido para o pacote que está a associar.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

SetBusyWindowButton()

`SetBusyWindowButton` vazio(*Text* BSTR)

Descrição

Esta função especifica o texto apresentado no botão Cancelar da janela ocupada.

Parâmetros

Nome	Descrição
Text	Uma cadeia que especifica o texto apresentado no botão Cancelar da janela ocupada. O valor predefinido é "Cancelar". Se especificar uma cadeia vazia, o botão ficará oculto. Independentemente do texto que especificar, o botão cancela ou fecha sempre a janela.

Valor de Retorno

Nenhum.

Tópicos Relacionados

SetBusyWindowMessage()

SetBusyWindowMode()

SetBusyWindowTitle()

ShowBusyWindow()

SetBusyWindowMessage()

SetBusyWindowMessage *vazio*(*Message* BSTR)

Descrição

Esta função especifica o texto apresentado na área de mensagem da janela ocupada.

Parâmetros

Nome	Descrição
Message	Uma cadeia que especifica o texto apresentado na área de mensagem da janela ocupada.

Valor de Retorno

Nenhum.

Tópicos Relacionados

SetBusyWindowButton()

SetBusyWindowMode()

SetBusyWindowTitle()

ShowBusyWindow()

SetBusyWindowMode()

SetBusyWindowMode *vazio*(*Mode* curto)

Descrição

Esta função determina a apresentação, ou não, da janela ocupada pelo QMF para Windows. A janela ocupada é útil para fornecer informações ao utilizador e para lhe permitir cancelar uma acção de base de dados pendente. As alterações produzem efeito da próxima vez que o QMF para Windows efectuar uma operação que faça com que a janela ocupada seja apresentada ou ocultada.

Parâmetros

Nome	Descrição
Mode	Especifica quando o QMF para Windows apresenta a janela ocupada:
Valor	Significado
0 (RSM_NEVER)	A janela não é apresentada. Este corresponde ao valor assumido.
1 (RSM_WHENBUSY)	A janela é apresentada quando o QMF para Windows estiver ocupado a comunicar com a base de dados. O QMF para Windows apresenta automaticamente esta janela conforme apropriado.
2 (RSM_CLIENTCONTROLLED)	A janela é apresentada depois de chamar ShowBusyWindow(TRUE), e depois de chamar ShowBusyWindow(FALSE). O cliente determina quando a janela é apresentada.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType() para obter informações de erro adicionais.

Tópicos Relacionados

SetBusyWindowButton()
SetBusyWindowMessage()
SetBusyWindowTitle()
SetParent()
ShowBusyWindow()

SetBusyWindowTitle()

SetBusyWindowTitle vazio (*Title* BSTR)

Descrição

Esta função especifica o texto apresentado na barra de título da janela ocupada.

Parâmetros

Nome	Descrição
Title	Uma cadeia que especifica o texto apresentado na barra de título da janela ocupada.

Valor de Retorno

Nenhum.

Tópicos Relacionados

SetBusyWindowButton()

SetBusyWindowMode()

SetBusyWindowMessage()

ShowBusyWindow()

SetGlobalVariable()

SetGlobalVariable curto(*Name* BSTR, *Value* BSTR)

Descrição

Esta função atribui um valor à variável global especificada, o qual fica disponível para utilização em consultas, formulários e procedimentos.

Parâmetros

Nome	Descrição
Name	Uma cadeia que contém o nome da variável que pretende definir.
Value	Uma cadeia que contém o valor que pretende atribuir à variável especificada.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType() para obter informações de erro adicionais.

SetHostVariable()

SetHostVariable curto(*QueryID* longo, VARIANT *Index*, VARIANT *Value*)

Descrição

Esta função atribui um valor à variável de sistema central especificada referenciada pela consulta. A consulta tem de ser estática e referenciar variáveis de sistema central (quer armazenadas com a consulta QMF quer criadas por AddHostVariable()). *Index* pode especificar quer o índice numérico da variável de sistema central quer o nome da mesma.

Parâmetros

Nome	Descrição
QueryID	O ID da consulta tal como é devolvido por InitializeStaticQuery().
Index	Um número (tipo de variante VT_I2) especificando o índice da variável de sistema central na consulta, ou uma cadeia (tipo de variante VT_BSTR) especificando o nome da mesma.
Value	O valor para a variável de sistema central. Para especificar um valor nulo, o tipo da variante deve ser definido como VT_EMPTY.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType() para obter informações de erro adicionais.

SetOption()

SetOption curto(*Mode* curto, VARIANT *Value*)

Descrição

Esta função define o valor da opção especificada no QMF para Windows. No caso de algumas opções, as alterações só produzem efeito quando o QMF para Windows for reiniciado. Em condições normais, só se reinicia o QMF para Windows depois de ter destruído todas as instâncias do objecto de API do QMF para Windows.

Parâmetros

Nome	Descrição
Option	Especifica a opção a definir:

Valor	Significado
0 (RSO_SERVER_DEFINITION_FILE)	Nome de ficheiro de definição de servidores.
1 (RSO_CPIC_DLL)	Nome de ficheiro da DLL Fornecedora de CPI-C.
2 (RSO_CPIC_TIMEOUT_WARNING)	Tempo de espera de aviso CPI-C (em segundos). Este limite não é usado para a API do QMF para Windows.
3 (RSO_CPIC_TIMEOUT_CANCEL)	Tempo de espera de cancelamento CPI-C (em segundos).

4 (RSO_TCP_TIMEOUT_WARNING)	Tempo de espera de aviso TCP (em segundos). Este limite não é usado para a API do QMF para Windows.
5 (RSO_TCP_TIMEOUT_CANCEL)	Tempo de espera de cancelamento TCP (em segundos).
6 (RSO_DISPLAY_NULLS_STRING)	A cadeia utilizada para apresentar valores nulos.
7 (RSO_ENTER_NULLS_STRING)	A cadeia utilizada para inserir valores nulos.
8 (RSO_ENTER_DEFAULTS_STRING)	A cadeia utilizada para inserir valores predefinidos.
9 (RSO_TRACE_FILE_1)	Nome do ficheiro de rastreio 1.
10 (RSO_TRACE_FILE_2)	Nome do ficheiro de rastreio 2.
11 (RSO_TCP_TRACE_LEVEL)	Nível de rastreio TCP.
12 (RSO_CPIC_TRACE_LEVEL)	Nível de rastreio CPI-C.
13 (RSO_DDM_TRACE_LEVEL)	Nível de rastreio DDM.
Nome	Descrição
Value	O valor para o qual definir a opção.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()` para obter informações de erro adicionais.

Tópicos Relacionados

`GetOption()`

SetParent()

`SetParent` curto(*ParentWnd* longo)

Descrição

Esta função define a janela ascendente para caixas de diálogo. Regra geral, quando o QMF para Windows apresenta uma caixa de diálogo (na janela ocupada ou na caixa de diálogo Informações de Utilizador), esta fica centrada e modal em relação à janela principal do QMF para Windows. Esta função permite-lhe forçar as caixas de diálogo do QMF para Windows a ficarem centradas e modais em relação à janela da sua aplicação cliente.

Parâmetros

Nome	Descrição
------	-----------

ParentWnd	O HWND da nova janela ascendente. Especifique NULL para utilizar a janela principal do QMF para Windows como ascendente.
-----------	--

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType() para obter informações de erro adicionais.

Tópicos Relacionados

ShowBusyWindow()

SetProcVariable()

SetProcVariable curto(*ProcID* longo, *Name* BSTR, *Value* BSTR)

Descrição

Esta função atribui um valor à variável especificada, o qual é substituído pela variável antes de executar o procedimento. Se o procedimento tiver uma ou mais variáveis, tem de se chamar esta função para definir os valores de variável antes de chamar RunProc().

Parâmetros

Nome	Descrição
ProcID	O ID do procedimento tal como é devolvido por InitializeProc().
Name	Uma cadeia que contém o nome da variável que pretende definir.
Value	Uma cadeia que contém o valor que pretende atribuir à variável especificada.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType() para obter informações de erro adicionais.

SetVariable()

SetVariable curto(*QueryID* longo, *Name* BSTR, *Value* BSTR)

Descrição

Esta função atribui um valor à variável especificada, o qual é substituído pela variável antes de executar a instrução SQL. Se a instrução SQL tiver uma ou mais variáveis, tem de se chamar esta função para definir os valores de variável antes de chamar Open() ou Execute().

Parâmetros

Nome	Descrição
QueryID	O ID da consulta tal como é devolvido por InitializeQuery().
Name	Uma cadeia que contém o nome da variável que pretende definir.
Value	Uma cadeia que contém o valor que pretende atribuir à variável especificada.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar GetLastErrorString() ou GetLastErrorType() para obter informações de erro adicionais.

ShowBusyWindow()

ShowBusyWindow *vazio*(*Show* BOOL)

Descrição

Esta função diz ao QMF para Windows para mostrar ou ocultar a janela ocupada. A janela ocupada é útil para fornecer informações ao utilizador e para lhe permitir cancelar uma acção de base de dados pendente. Esta função só produz efeito se chamar SetBusyWindowMode() com um modo de RSM_CLIENTCONTROLLED. Se definir uma janela ascendente chamando SetParent(), esta será modal em relação à janela especificada.

Parâmetros

Nome	Descrição
Show	Um valor diferente de zero mostra a janela ocupada; zero oculta a mesma. Se for diferente de zero, a janela ocupada é apresentada até chamar ShowBusyWindow() com <i>Show</i> definido como zero.

Valor de Retorno

Nenhum.

StartBind()

StartBind curto(*CollectionName* BSTR, *PackageName* BSTR, *ConsistencyToken* BSTR)

Descrição

Esta função começa o processo de associação de um pacote na base de dados.

Parâmetros

Nome	Descrição
CollectionName	O ID de coleção pretendido para o pacote.
PackageName	O nome pretendido para o pacote.
ConsistencyToken	Uma cadeia com 16 caracteres de comprimento contendo a representação hexadecimal de um símbolo de oito bytes utilizado para garantir coerência entre o pacote associado na base de dados e uma aplicação que utilize tal pacote. Quando se executa uma secção no pacote também tem de se fornecer este mesmo valor.

Valor de Retorno

Zero se tiver êxito, diferente de zero se não tiver êxito. Se o valor de retorno for diferente de zero, pode-se chamar `GetLastErrorString()` ou `GetLastErrorType()`, `GetLastSQLCode()`, `GetLastSQLError()`, ou `GetLastSQLState()` para obter informações de erro adicionais.

Tópicos Relacionados

`EndBind()`
`CancelBind()`

Apêndice. Informações

Estas informações aplicam-se a produtos e serviços disponíveis nos Estados Unidos da América. É possível que a IBM não venha a disponibilizar noutros países os produtos, serviços ou módulos referidos neste documento. Consulte o seu representante de vendas IBM local para obter informações sobre os produtos e serviços actualmente disponíveis na sua área. Quaisquer referências a programas autorizados da IBM ou a outros produtos ou serviços IBM, não significam que apenas esses programas autorizados, produtos ou serviços da IBM possam ser utilizados. Qualquer outro produto, programa ou serviço funcionalmente equivalente poderá ser utilizado em substituição daqueles, desde que não infrinja quaisquer direitos de propriedade industrial da IBM. Contudo, o utilizador será considerado responsável pela avaliação e verificação do funcionamento de qualquer produto, programa ou serviço que não seja da IBM.

Neste documento podem ser feitas referências a patentes ou a pedidos de patente pendentes. O facto de estas informações lhe serem fornecidas não lhe confere quaisquer direitos sobre essas patentes. Todos os pedidos de informações sobre licenças deverão ser endereçados a:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10594-1785
E.U.A.

Para obter informações sobre licenças relativas a conjunto de caracteres de duplo byte (DBCS) deverá contactar o IBM Intellectual Property Department do seu país ou enviar o seu pedido, por escrito, para:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

O parágrafo seguinte não se aplica ao Reino Unido nem a qualquer outro país onde estas cláusulas sejam incompatíveis com a lei local: A INTERNATIONAL BUSINESS MACHINES CORPORATION FORNECE ESTA PUBLICAÇÃO “TAL COMO ESTÁ” SEM QUALQUER GARANTIA, QUER EXPLÍCITA QUER IMPLÍCITA, INCLUINDO, MAS NÃO SE LIMITANDO, ÀS GARANTIAS IMPLÍCITAS DE NÃO INFRACÇÃO, COMERCIALIZAÇÃO OU ADEQUAÇÃO A UM DETERMINADO FIM. Alguns Estados não

permitem a exclusão de garantias, quer explícitas quer implícitas, em determinadas transacções; esta declaração pode, portanto, não se aplicar ao seu caso.

É possível que esta publicação contenha imprecisões técnicas ou erros de tipografia. A IBM permite-se fazer alterações periódicas às informações aqui contidas; essas alterações serão incluídas nas posteriores edições desta publicação. A IBM permite-se fazer aperfeiçoamentos e/ou alterações aos produto(s) e/ou programa(s) descritos nesta publicação em qualquer momento, sem aviso prévio.

Quaisquer referências, nesta publicação, a sites da Web que não sejam da IBM são fornecidas para conveniência e não deverão nunca servir como aprovação desses sites da Web. Os materiais nesses sites da Web não são parte dos materiais para este produto da IBM e a utilização desses sites da Web corre por seu próprio risco.

Os possuidores de licenças deste programa que pretendam obter informações sobre o mesmo com o objectivo de permitir: (i) a troca de informações entre programas criados independentemente e outros programas (incluindo este) e (ii) a utilização recíproca das informações que tenham sido trocadas, deverão contactar a IBM.

IBM Corporation
J74/G4
555 Bailey Avenue
P.O. Box 49023
San Jose, CA 95161-9023
E.U.A.

Tais informações poderão estar disponíveis, sujeitas aos termos e às condições adequadas, incluindo, nalguns casos, o pagamento de uma taxa.

O programa autorizado descrito nestas informações, bem como todos os respectivos materiais licenciados disponíveis, são fornecidos pela IBM nos termos do Acordo de Cliente IBM ou de qualquer outro acordo equivalente existente entre as partes.

Os dados relativos ao rendimento aqui referidos foram obtidos num ambiente controlado. Consequentemente, os resultados obtidos em qualquer outro ambiente podem variar significativamente. É possível que algumas das medições efectuadas tenham sido efectuadas em sistemas de nível de desenvolvimento, não existindo qualquer garantia de que as mesmas sejam idênticas às geralmente obtidas em sistemas de comercialização geral. Além disso, é possível que algumas medições correspondam a estimativas, obtidas

por extrapolação. Os resultados reais podem variar. Os utilizadores do presente documento deverão verificar os dados aplicáveis ao seu ambiente específico.

As informações relativas a produtos que não sejam da IBM foram obtidas junto de fornecedores dos referidos produtos, respectivos comunicados ou de outras fontes de divulgação ao público. A IBM não efectuou quaisquer testes aos referidos produtos e não pode, portanto, confirmar a exactidão do rendimento, compatibilidade ou quaisquer outras afirmações inerentes a produtos que não sejam da IBM. As questões relativas à capacidade de produtos que não sejam da IBM deverão ser endereçadas aos fornecedores dos referidos produtos.

Todas as afirmações relativas às directivas ou tendências futuras da IBM estão sujeitas a alterações ou descontinuação sem aviso prévio, representando apenas metas e objectivos.

Todos os preços de produtos da IBM são preços de revenda sugeridos pela IBM, actuais e estão sujeitos a alterações sem aviso prévio. Os preços dos fornecedores podem variar.

Estas informações são apenas para fins de planeamento. As informações incluídas no presente documento estão sujeitas a alterações, anteriores à disponibilização dos produtos nelas descritos.

Estas informações contêm exemplos de dados e de relatórios utilizados em operações comerciais no dia-a-dia. Para os ilustrar o melhor possível, os exemplos incluem nomes de indivíduos, firmas, marcas e produtos. Todos estes nomes são fictícios e qualquer semelhança com nomes e moradas reais é mera coincidência.

LICENÇA DE DIREITOS DE AUTOR:

Estas informações contêm programas de aplicação exemplo em linguagem de origem, que ilustram as técnicas de programação em diversas plataformas operativas. O utilizador está autorizado a copiar, modificar ou distribuir os referidos programas exemplo em qualquer formato, sem obrigação de pagamento de qualquer encargo à IBM, para efeitos de desenvolvimento, utilização, marketing ou distribuição de programas de aplicação em conformidade com a interface de programação de aplicações da plataforma operacional para a qual os programas exemplo foram desenvolvidos. Estes exemplos não foram testados de forma exaustiva, sob todas as condições possíveis. Consequentemente, a IBM não pode garantir ou confirmar a fiabilidade, adequabilidade ou funcionamento dos referidos programas.

A visualização em linha das actuais informações poderá originar a não visualização de fotografias e ilustrações a cor.

Marcas Comerciais

Os termos seguintes são marcas comerciais da IBM Corporation nos Estados Unidos e/ou noutros países:

ACF/VTAM	IBMLink
Advanced Peer-to-Peer Networking	IMS
AIX	Language Environment
AIX/6000	MVS/ESA
AS/400	MVS/XA
CICS	OfficeVision/VM
CICS/ESA	OS/2
CICS/MVS	OS/390
CICS/VSE	PL/I
COBOL/370	PROFS
DATABASE 2	QMF
DataJoiner	RACF
DB2	S/390
DB2 Universal Database	SQL/DS
Distributed Relational Database Architecture	Virtual Machine/Enterprise Systems Architecture
DRDA	Visual Basic
DXT	VM/XA
GDDM	VM/ESA
IBM	VSE/ESA
	VTAM

Java ou todas as marcas comerciais e logotipos com base em Java e Solaris são marcas comerciais da Sun Microsystems, Inc. nos Estados Unidos, outros países ou em ambos.

Lotus e 1-2-3 são marcas comerciais da Lotus Development Corporation nos Estados Unidos, outros países ou em ambos.

Microsoft, Windows e Windows NT são marcas comerciais ou marcas registadas da Microsoft Corporation.

Outros nomes de empresas, produtos e serviços, eventualmente identificados por dois asteriscos (**), podem ser marcas comerciais ou marcas de serviços de terceiros.

Índice Remissivo

A

abrir

- consultas orientadas no servidor de base de dados 24
- consultas SQL no servidor de base de dados 14
- ficheiros de consulta orientada 24
- ficheiros SQL guardados 13
- formulários guardados no servidor de base de dados 38
- procedimentos na base de dados 43

AddDecimalHostVariable() 73

AddHostVariable() 73

adicionar

- condições de linha 21
- condições de ordenação 20

adicionar objectos a listas 46

adicionar uma linha

- editor de tabelas 58

agrupar resultados de consulta 29, 51

alterar palavras-passe 3

alterar uma linha

- editor de tabelas 58

aplicações exemplo 64

apresentar objectos 46

B

barra de ferramentas

- adicionar botões 6
- mover botões 6
- personalizar 6
- remover botões 7

base de dados

- segurança 2

BindDecimalHostVariable() 75

BindHostVariable() 75

BindSection() 76

Bloquear chamadas 72

botões de acção

- consulta orientada 17

C

cadeias de contabilização 4

cálculos

- formulários 33

CancelBind() 77

ChangePassword() 77

ClearList() 78

Close() 78

colunas

- adicionar a consultas orientadas 19
- formulários 33

colunas, reordenar 28, 50

colunas, seleccionar 27

começando por (condição de linha) 21

Commit() 79

CompleteQuery() 79

complexas

- construir complexas 19

complexas, consultas

- construir 19

condições

- formulários 33

condições de linha

- adicionar 21
- começando por 21
- contendo 21
- entre 21
- igual a 20
- maior que 21
- maior que ou igual a 21
- menor que 21
- menor que ou igual a 21
- nulo 21
- terminando por 21
- utilizar 20

condições de ordenação

- adicionar 20
- utilizar 20

consulta orientada

- botões de acção 17

consultas

- construir consultas 19

consultas estáticas

- criar 53
- executar 55
- utilizar variáveis de substituição 53

consultas múltiplas

- apresentar simultaneamente 10

consultas SQL

- abrir ficheiros guardados 13
- abrir um novo documento 9

consultas SQL (*continuação*)

- executar num servidor de base de dados 9
- guardar em ficheiros 13
- guardar no servidor de base de dados 14
- imprimir 15
- ver antes 15

contendo (condição de linha) 21

converter formatação num

- formulário 29, 51

CopyToClipboard() 80

Correio da Internet 63

criar

- consultas estáticas 53

criar ficheiros de trabalho 49

criar um procedimento linear 41

D

DeleteQMFObject() 81

detalhes

- formulários 33

documentos de consulta

- múltipla 10

E

editar objecto 46

editor de tabelas 57

- adicionar uma linha 58

- alterar uma linha 58

- procurar linhas 57

EndBind() 81

entre (condição de linha) 21

Enviar Para 63

Excel

- suplemento 64

executar

- consulta SQL num servidor de base de dados 9
- consultas estáticas 55
- orientadas, consultas 18

executar objecto 46

Execute() 82

ExecuteEx() 82

ExecuteStoredProcedure() 83

ExecuteStoredProcedureEx() 84

Export() 86

exportar

- relatórios 39

- exportar dados
 - para ficheiros 61
 - para outras tabelas 63
 - para um servidor de base de dados 63
- ExportForm() 88
- ExportReport() 88
- extrair consultas
 - criar 11
- extrair objecto 46
- F**
- FastSaveData() 90
- FetchNextRow() 91
- FetchNextRowEx() 92
- FetchNextRows() 92
- FetchNextRowsEx() 94
- ficheiros
 - exportar dados para 61
- ficheiros de trabalho, criar 49
- final
 - formulários 33
- FlushQMFCache() 94
- formatar resultados de consulta
 - numéricos 29, 51
- formulário
 - principal 33
- formulários
 - abrir um ficheiro guardado 37, 42
 - cálculos 33
 - colunas 33
 - condições 33
 - detalhes 33
 - elaborar um relatório 34
 - final 33
 - guardar em ficheiros 37, 42
 - guardar no servidor de base de dados 37, 42
 - HTML 34
 - opções 34
 - página 34
 - quebras 33
- Formulários de DB2 59
- G**
- GetColumnCount() 95
- GetColumnDataValue() 95
- GetColumnHeader() 96
- GetColumnHeaderEx() 96
- GetColumnHeadings() 97
- GetColumnValue() 98
- GetColumnValueEx() 98
- GetDefaultServerName() 99
- GetGlobalVariable() 99
- GetHostVariableNames() 99
- GetHostVariableTypes() 100
- GetLastErrorString() 100
- GetLastErrorType() 101
- GetLastSQLCode() 102
- GetLastSQLError() 103
- GetLastSQLState() 104
- GetOption() 104
- GetOptionEx() 106
- GetProcText() 106
- GetProcVariables() 107
- GetQMFOBJECTInfo() 108
- GetQMFOBJECTInfoEx() 109
- GetQMFOBJECTList() 111
- GetQMFOBJECTListEx() 112
- GetQMFPProcText() 113
- GetQMFPQueryText() 113
- GetQueryText() 114
- GetQueryVerb() 114
- GetResourceLimit() 115
- GetResourceLimitEx() 119
- GetRowCount() 120
- GetServerList() 120
- GetServerListEx() 121
- GetStoredProcedureResultSets() 122
- GetVariables() 122
- GetVariablesEx() 123
- guardar
 - consultas SQL em ficheiros 13
 - consultas orientadas como ficheiros 23
 - consultas orientadas no servidor de base de dados 24
 - consultas SQL no servidor de base de dados 14
 - formulários em ficheiros 37, 42
 - formulários no servidor de base de dados 37, 42
- guardar resultados de consulta 30, 51
- guardar resultados de consulta em ficheiros 30, 52
- H**
- HTML
 - formulários 34
- I**
- igual a (condição de linha) 20
- imprimir
 - consultas SQL 15
 - procedimento 44
 - relatórios 38
- imprimir resultados de consulta 30, 52
- Informações 145
- iniciar sessão 2
- InitializeProc() 124
- InitializeQuery() 124
- InitializeServer() 125
- InitializeStaticQuery() 126
- IsStatic() 127
- J**
- junção, condições de
 - criar em consultas orientadas 22
- L**
- Ligar à base de dados 72
- linhas, seleccionar 27
- listar
 - objectos 45
- listas
 - abrir ficheiros guardados 47
- listas, adicionar objectos 46
- listas, remover objectos 47
- M**
- maior que (condição de linha) 21
- maior que ou igual a (condição de linha) 21
- menor que (condição de linha) 21
- menor que ou igual a (condição de linha) 21
- múltiplas, tabelas
 - em consultas orientadas 22
- N**
- Não É (operador de condição de linha) 21
- nova
 - consulta SQL 9
 - extrair consultas 11
- novas
 - orientadas, consultas 17
- nulo (condição de linha) 21
- O**
- objectos
 - listar 45
- opções
 - formulários 34
- Open() 127
- operador de condição de linha
 - Não É 21
 - Sim É 21
- ordenar resultados de consulta 28, 49
- orientadas, consultas
 - abrir ficheiros guardados 24
 - adicionar colunas 19
 - adicionar tabelas 18
 - converter em SQL 23
 - criar 17

orientadas, consultas (*continuação*)
criar condições de junção 22
executar 18
guardar como ficheiros 23
guardar no servidor de base de dados 24
utilizar SQL em 22
utilizar tabelas múltiplas 22
utilizar variáveis de substituição 23
ver SQL 23

P

página
formulários 34
palavras-passe
corrigir 3
pré-visualizar
consultas impressas 15
procedimento impresso 43
relatórios 34
pré-visualizar resultados de consulta 30, 52
Prepare() 128
principal
formulários 33
PrintReport() 129
procedimento
imprimir 44
procedimento com lógica 41
procedimento REXX 41
procurar
editor de tabelas 57

Q

quebras
formulários 33

R

redimensionar colunas e linhas 27
Referência API 73
regulação 4
ReinitializeServer() 129
relatórios
elaborar um relatório usando formulários 34
exportar 39
imprimir 38
pré-visualizar 34
remover objectos de listas 47
reordenar colunas 28, 50
resultados de consulta, agrupar 29, 51
resultados de consulta, formatar 29, 51

resultados de consulta, guardar 30, 51
resultados de consulta, guardar em ficheiros 30, 52
resultados de consulta,
imprimir 30, 52
resultados de consulta, ordenar 28, 49
resultados de consulta,
pré-visualizar 30, 52
resultados de consulta, resumir 29, 51
resumir resultados de consulta 29, 51
Rollback() 129
RunProc() 130

S

SaveData() 130
SaveQMFPProc() 133
SaveQMFPQuery() 133
Selecionar colunas e linhas 27
servidor
definir 1
servidor de base de dados
exportar dados para 63
SetBindOption() 134
SetBindOwner() 136
SetBusyWindowButton() 136
SetBusyWindowMessage() 137
SetBusyWindowMode() 137
SetBusyWindowTitle() 138
SetGlobalVariable() 139
SetHostVariable() 139
SetOption() 140
SetParent() 141
SetProcVariable() 142
SetVariable() 142
ShowBusyWindow() 143
Sim É (operador de condição de linha) 21
SQL
utilizar em consultas orientadas 22
StartBind() 143
suplemento
Excel 64

T

tabelas
adicionar a consultas orientadas 18
exportar dados para 63
terminando por 21
tipos de letra
apresentação da consulta 10

tipos de letra (*continuação*)
apresentação de resultados 29, 50

V

variáveis de sistema central
utilizar em consultas estáticas 53
variáveis de substituição
em consultas SQL 12
executar consultas SQL com 12
substituir por variáveis de substituição 53
utilizar em consultas estáticas 53
utilizar em consultas orientadas 23
ver
resultados 9
SQL 9
SQL em consultas orientadas 23
ver antes
orientadas, consultas 25
vista resultados 9

Comentários do Leitor

Query Management Facility
Iniciação ao QMF para Windows
Versão 7

Número da Publicação SC17-5378-00

Gostaríamos de conhecer a sua opinião sobre esta publicação. Se desejar, envie-nos os seus comentários sobre erros ou omissões, exactidão, organização, temas tratados, ou limitações deste manual. Os comentários que enviar deverão limitar-se às informações contidas neste manual e ao modo como são apresentadas.

Para perguntas técnicas e informações sobre produtos e preços, por favor contacte o escritório da IBM, um parceiro de negócios IBM, ou um concessionário autorizado.

Para questões genéricas, por favor contacte a Companhia IBM Portuguesa (Telefone 7915000).

Os comentários que enviar para a IBM, garantem à IBM um direito não exclusivo de os utilizar ou distribuir do modo que a IBM julgar adequado, sem contudo incorrer em qualquer obrigação para consigo.

Comentários:

Obrigado pelo seu apoio.

Para submeter os seus comentários:

- Envie os seus comentários para o endereço que está no reverso deste formulário.
- Envie os seus comentários via e-mail para: <http://www.dss.ibm.com>

Se desejar uma resposta da IBM, por favor preencha as seguintes informações:

Nome

Morada

Empresa

Número de telefone

Endereço E-mail



IBM CORPORATION
Department BWE/H3
P.O. Box 49023
San Jose, CA 95161-9023
U.S.A.



Número de Ficheiro:

Número do Programa: 5675-DB2
5697-F42
5697-G24
5697-G22
5648-D35
5697-G23

Impresso em Portugal

SC17-5378-00



Spine information:



QMF

Iniciação ao QMF para Windows

Versão 7