

Bases de Dados

(2000/2001)

Considere o esquema relacional seguinte:

Departamento(DepNum, Nome, Local)

Empregado(EmpNum, Nome, Categoria, Salario, DepNum)

Projecto(ProjNum, Designacao, Fundos)

Atribuicao(EmpNum, ProjNum, Funcao)

1. Criar a base de dados *Projecto.mdb*:

{ *Ficheiro* → *Nova base de dados...* ou CTRL+O }

Crie a base de dados com o nome *Projecto.mdb*.

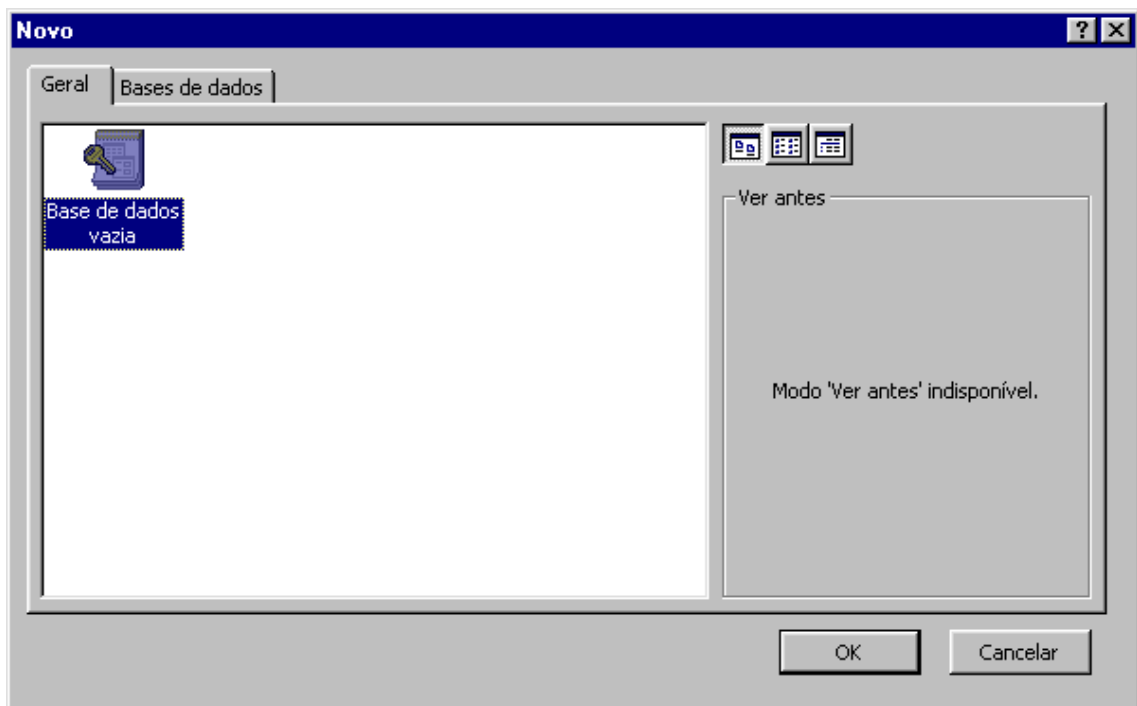


Fig. 1 Diálogo para criar uma base de dados.

2. Criar as tabelas correspondentes ao esquema relacional anterior:

{Selecione o separador Tabelas, pressione o botão Novo e escolha a opção Vista de estrutura. }

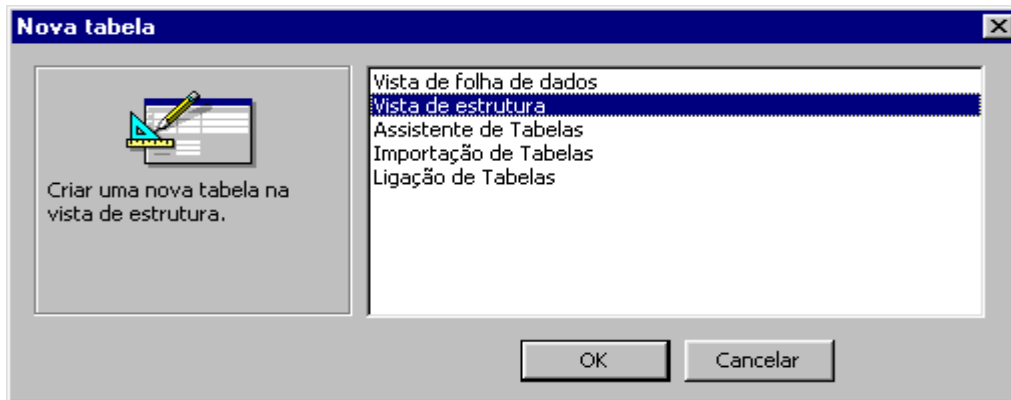


Fig. 2 Criar uma nova tabela.

Crie as quatro tabelas usando os seguintes elementos:

Departamento(DepNum, Nome, Local)

DepNum → Número (Chave primária)
Nome → Texto 30 Chars
Local → Texto 30 Chars

Empregado(EmpNum, Nome, Categoria, Salario, DepNum)

EmpNum → Número (Chave primária)
Nome → Texto 50 Chars
Categoria → Texto 30 Chars
Salario → Número (Formatado como moeda)
DepNum → Número (Chave forasteira)

Projecto(ProjNum, Designacao, Fundos)

ProjNum → Número (Chave primária)
Designacao → Texto 30 Chars
Fundos → Número (Formatado como moeda)

Atribuicao(EmpNum,ProjNum, Funcao)

EmpNum → Número (Chave primária)
ProjNum → Número (Chave primária)
Funcao → Texto 30 Chars

3. Estabelecer as chaves estrangeiras (ou forasteiras):

{ Ferramentas → Relações...

Adicione todas as tabelas ao seu modelo. }

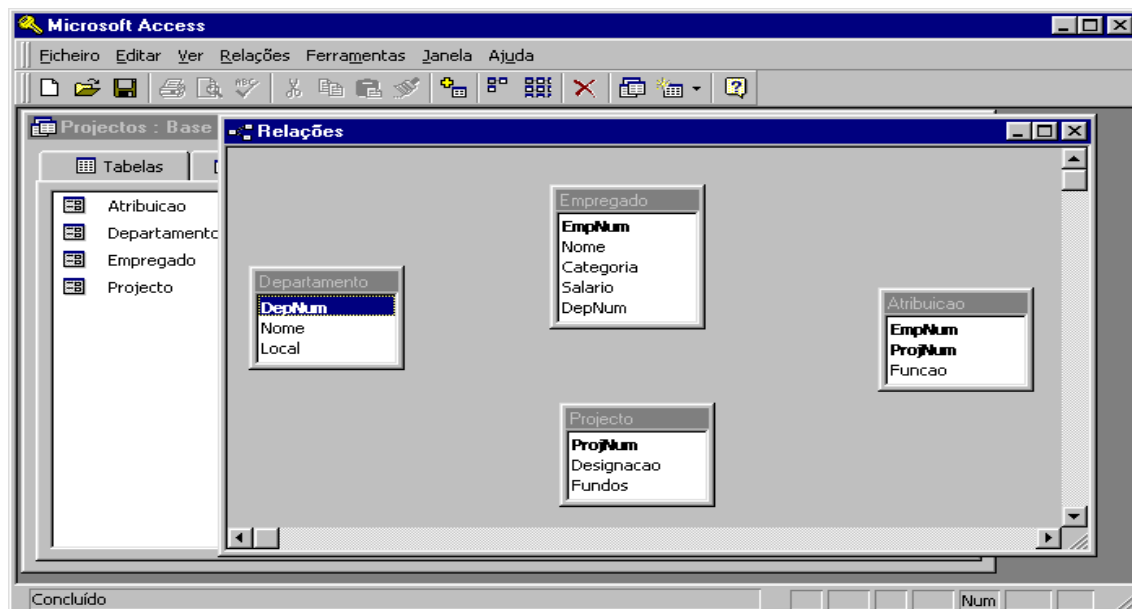


Fig. 3 Tabelas a usar no diagrama de relacionamentos.

Chave estrangeira em Empregado:

"Todo o empregado pertence a um único departamento mas um departamento pode ter vários empregados."

O atributo DepNum na tabela Empregado corresponde a um DepNum na tabela Departamento. DepNum é chave estrangeira na tabela Empregado.

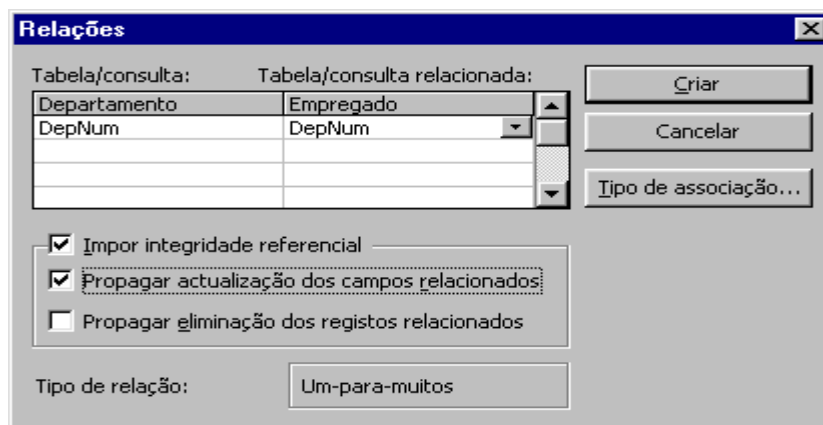


Fig. 4 Integridade referencial.

Para estabelecer a integridade referencial proceda da forma seguinte:

No esquema de relações seleccione o atributo DepNum da tabela Departamento e arraste-o e largue-o sobre a tabela Empregado.

Esta acção faz aparecer o diálogo da Fig. 4. Neste diálogo seleccione o atributo DepNum da tabela Empregado.

Finalmente, seleccione *Impor integridade referencial* e *Propagar actualização dos campos relacionados* e crie o relacionamento.

Proceda de igual forma para os relacionamentos da tabela Atribuição.

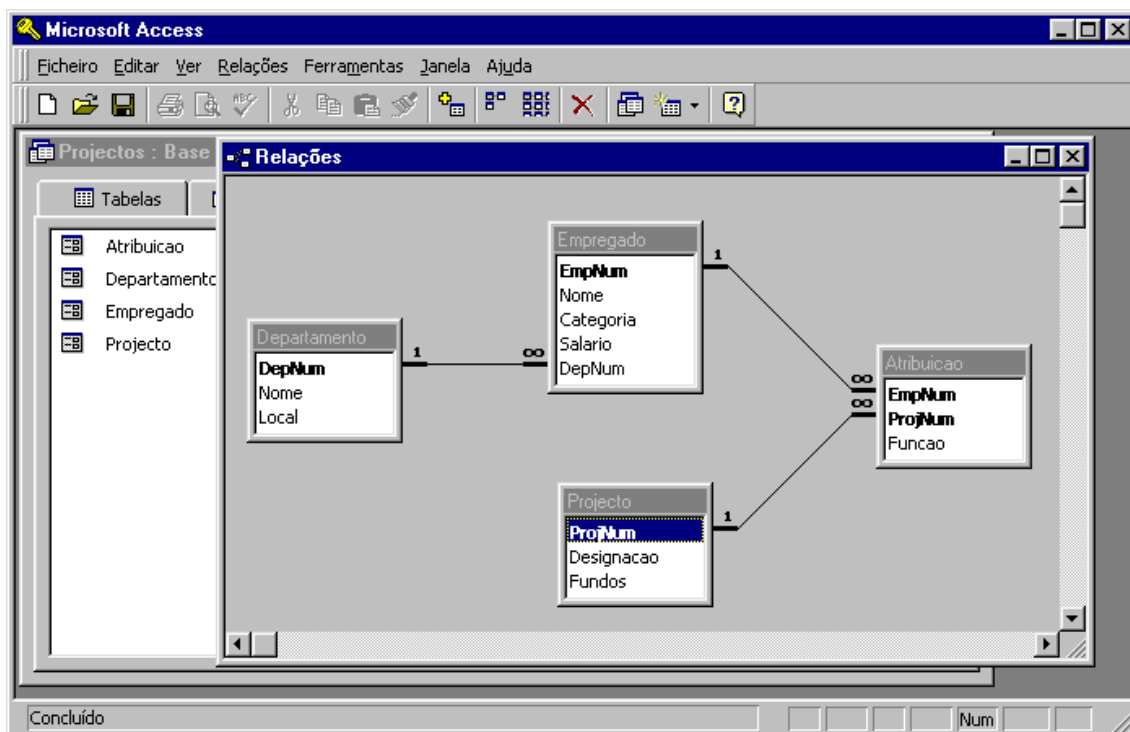


Fig. 5 Vista global sobre o diagrama de relacionamentos.

4. Criar formulários:

{Posicione-se sobre o separador Tabelas e seleccione a tabela para a qual pretende construir o formulário. De seguida, seleccione a opção *Formulário automático* do menu *Inserir*}

Crie formulários para as tabelas do esquema relacional. Dê nomes apropriados para os formulários (Projecto, Departamento,...).

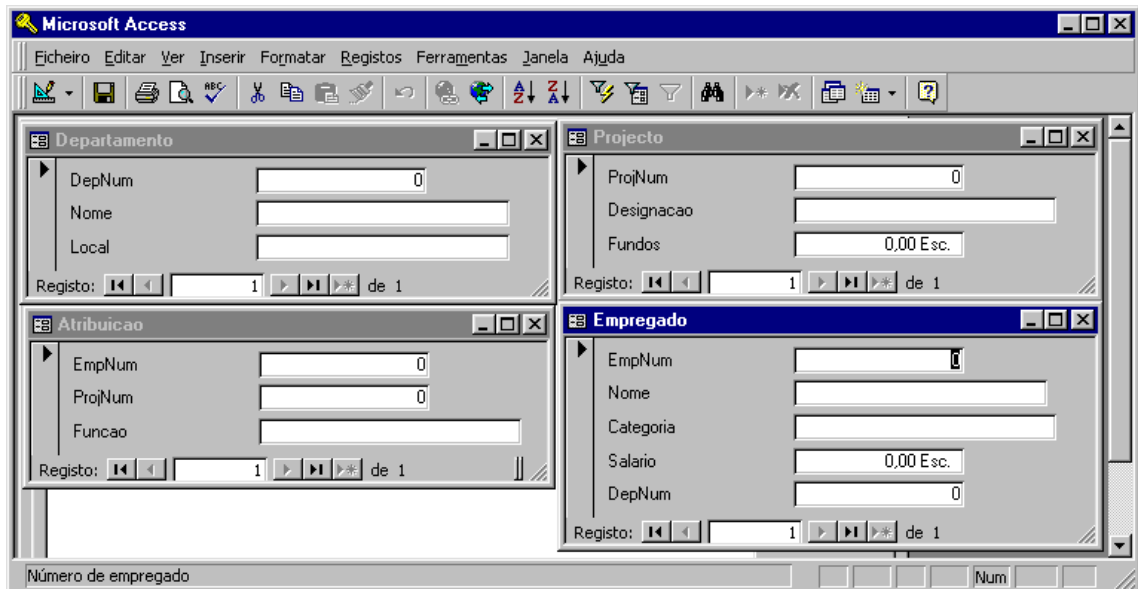


Fig. 6 Criar formulários (de modo automático).

5. Retoques nos formulários:

Seleccione o formulário *Atribuição*.



The screenshot shows a window titled 'Atribuição' with three text input fields: 'EmpNum' (containing '0'), 'ProjNum' (containing '0'), and 'Funcao'. At the bottom, there is a record navigation bar with the text 'Registo: 1 de 1'.

Fig. 7 Formulário antes dos retoques.

Altere as etiquetas para nomes mais apropriados.

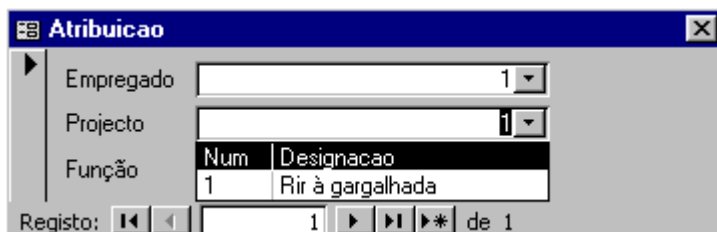
Active a caixa de texto correspondente ao número do empregado. Altere o tipo de controlo de modo a seleccionar empregados da tabela Empregado. Para tal proceda do modo seguinte:

"Click" com o botão direito do rato sobre a caixa de texto e seleccione a opção *Alterar para caixa de combinação*.

Na lista de propriedades altere as propriedades seguintes:

<i>Origem da linha</i>	SELECT E.EmpNum as Num, E.Nome FROM Empregado E;
<i>Número de colunas</i>	2
<i>Títulos de coluna</i>	Sim
<i>Largura das colunas</i>	1;3;

Proceda de modo análogo para o número de projecto.



The screenshot shows the 'Atribuição' form after modifications. The 'Empregado' field is now a dropdown menu with '1' selected. The 'Projecto' field is also a dropdown menu with '1' selected. The 'Função' field is a table with two columns: 'Num' and 'Designacao'. The first row contains '1' and 'Rir à gargalhada'. At the bottom, the record navigation bar shows 'Registo: 1 de 1'.

Fig. 8 Formulário após retoques.

6. Experiências diversas

O formulário *Atribuição*, obtido no ponto anterior, permite seleccionar dados sempre válidos, pois os números de empregado e projecto são obtidos das respectivas tabelas. No entanto, nem sempre assim acontece; de facto, podia tentar-se inserir quaisquer valores nestes campos. Contudo, o sistema só aceitaria valores que existissem nas tabelas correspondentes, pois esses campos foram marcados como sendo chaves estrangeiras.

Algumas questões:

1. Pode-se alterar o atributo EmpNum, da tabela Empregado, de um empregado que participe num projecto? E nesse caso o que acontece ao EmpNum na tabela Atribuição?
2. Os empregados que participam em Projecto(s) podem ser removidos da tabela Empregado?

Para se responder à questão 1, i.e., para se ver a integridade referencial em acção (definida no ponto 3), sugere-se uma pequena experiência:

Abrir a tabela Empregado e alterar o atributo EmpNum. De seguida, abrir a tabela Atribuição e verificar que os correspondentes tuplos, onde aparecia o número de empregado alterado, foram modificados para reflectir a nova realidade.

Notar que no ponto 3 (definir relações) foi criado um relacionamento entre Empregado e Atribuição e informou-se o sistema que devia *impor a integridade referencial e propagar a actualização dos campos relacionados*.

Faça algumas experiências, veja os resultados e tire as suas conclusões.

7. Experiências com SQL

Considere as ocorrências seguintes:

Departamento:

DepNum	Nome	Local
1	Camarote	Camarate
2	Informática	Covilhã
3	Produção	Guarda
4	Comercial	Lisboa

Empregado:

EmpNum	Nome	Categoria	Salario	DepNum
1	José	Artista	230.000,00 Esc.	1
2	Artur	Programador	250.000,00 Esc.	2
3	Carlos	Analista	300.000,00 Esc.	2
4	Zezinho	Artes gráficas	150.000,00 Esc.	1
5	Abílio	Chefe secção	100.000,00 Esc.	3
6	António	Design	100.000,00 Esc.	3
7	Daniela	Modelo	80.000,00 Esc.	3
8	Margarida	Director	350.000,00 Esc.	4
9	Fazenda	Patrão	600.000,00 Esc.	4
10	Marília	Secretária	85.000,00 Esc.	4

Projecto:

ProjNum	Designacao	Fundos
1	Rir à gargalhada	60.000,00 Esc.
2	BD do IPG	5.000.000,00 Esc.
3	Sapataria da Beira	3.000.000,00 Esc.

Atribuição:

EmpNum	ProjNum	Funcao
1	1	Animador
2	2	Colaborador
3	2	Coordenador
3	3	Colaborador
5	3	Coordenador
6	3	Colaborador
7	3	Colaborador

Construa as consultas seguintes:

a) Projecção

"Obtenha uma listagem com os nomes e categorias dos empregados"

```
SELECT Nome, Categoria  
FROM Empregado
```

Conjunto resultado:

Nome	Categoria
José	Artista
Artur	Programador
Carlos	Analista
Zezinho	Artes gráficas
Abílio	Chefe secção
António	Design
Daniela	Modelo
Margarida	Director
Fazenda	Patrão
Marília	Secretária

b) Restrição

"Obtenha uma listagem com todos os atributos dos empregados programadores"

```
SELECT *  
FROM Empregado  
WHERE Categoria = 'Programador'
```

Conjunto resultado:

EmpNum	Nome	Categoria	Salario	DepNum
2	Artur	Programador	250.000,00 Esc.	2

c) Restrição e Projecção

"Obtenha uma listagem com o nome dos empregados programadores"

```
SELECT Nome  
FROM Empregado  
WHERE Categoria = 'Programador'
```

Conjunto resultado:

Nome
Artur

d) União de projecções

"Obtenha uma listagem com os números dos empregados que já desempenharam as funções de coordenador ou de colaborador em algum projecto"

```
SELECT EmpNum  
FROM Atribuicao  
WHERE Funcao = 'Coordenador'
```

UNION

```
SELECT EmpNum  
FROM Atribuicao  
WHERE Funcao = 'Colaborador'
```

Conjunto resultado:

EmpNum
2
3
5
6
7

e) Intersecção

Operação indisponível no MS ACCESS.

"Obtenha uma listagem com os números dos empregados que já desempenharam as funções de coordenador e de colaborador em algum projecto"

```
SELECT EmpNum  
FROM Atribuicao  
WHERE Funcao = 'Coordenador'
```

INTERSECT

```
SELECT EmpNum  
FROM Atribuicao  
WHERE Funcao = 'Colaborador'
```

Conjunto resultado:

EmpNum
3

f) Diferença

Operação indisponível no MS ACCESS.

"Obtenha uma listagem com os números dos empregados que desempenharam sempre a função de coordenador nos projectos em que participaram"

```
SELECT EmpNum  
FROM Atribuicao  
WHERE Funcao = 'Coordenador'
```

MINUS

```
SELECT EmpNum  
FROM Atribuicao  
WHERE Funcao <> 'Coordenador'
```

Conjunto resultado:

EmpNum
5

g) Produto cartesiano

"Difícilmente seria pedido um produto cartesiano. No entanto, se este fosse necessário, também poderia ser obtido. "

SELECT *
FROM Empregado , Departamento
Conjunto resultado:

Emp Num	Empregado . Nome	Categoria	Salario	Empregado . DepNum	Departamento . DepNum	Departamento . Nome	Local
1	José	Artista	230.000,00 Esc.	1	1	Camarote	Camarate
1	José	Artista	230.000,00 Esc.	1	2	Informática	Covilhã
1	José	Artista	230.000,00 Esc.	1	3	Produção	Guarda
1	José	Artista	230.000,00 Esc.	1	4	Comercial	Lisboa
2	Artur	Programador	250.000,00 Esc.	2	1	Camarote	Camarate
2	Artur	Programador	250.000,00 Esc.	2	2	Informática	Covilhã
2	Artur	Programador	250.000,00 Esc.	2	3	Produção	Guarda
2	Artur	Programador	250.000,00 Esc.	2	4	Comercial	Lisboa
3	Carlos	Analista	300.000,00 Esc.	2	1	Camarote	Camarate
3	Carlos	Analista	300.000,00 Esc.	2	2	Informática	Covilhã
3	Carlos	Analista	300.000,00 Esc.	2	3	Produção	Guarda
3	Carlos	Analista	300.000,00 Esc.	2	4	Comercial	Lisboa
4	Zezinho	Artes gráficas	150.000,00 Esc.	1	1	Camarote	Camarate
4	Zezinho	Artes gráficas	150.000,00 Esc.	1	2	Informática	Covilhã
4	Zezinho	Artes gráficas	150.000,00 Esc.	1	3	Produção	Guarda
4	Zezinho	Artes gráficas	150.000,00 Esc.	1	4	Comercial	Lisboa
5	Abílio	Chefe secção	100.000,00 Esc.	3	1	Camarote	Camarate
5	Abílio	Chefe secção	100.000,00 Esc.	3	2	Informática	Covilhã
5	Abílio	Chefe secção	100.000,00 Esc.	3	3	Produção	Guarda
5	Abílio	Chefe secção	100.000,00 Esc.	3	4	Comercial	Lisboa
6	António	Design	100.000,00 Esc.	3	1	Camarote	Camarate
6	António	Design	100.000,00 Esc.	3	2	Informática	Covilhã
6	António	Design	100.000,00 Esc.	3	3	Produção	Guarda
6	António	Design	100.000,00 Esc.	3	4	Comercial	Lisboa
7	Daniela	Modelo	80.000,00 Esc.	3	1	Camarote	Camarate
7	Daniela	Modelo	80.000,00 Esc.	3	2	Informática	Covilhã
7	Daniela	Modelo	80.000,00 Esc.	3	3	Produção	Guarda
7	Daniela	Modelo	80.000,00 Esc.	3	4	Comercial	Lisboa
8	Margarida	Director	350.000,00 Esc.	4	1	Camarote	Camarate
8	Margarida	Director	350.000,00 Esc.	4	2	Informática	Covilhã
8	Margarida	Director	350.000,00 Esc.	4	3	Produção	Guarda
8	Margarida	Director	350.000,00 Esc.	4	4	Comercial	Lisboa
9	Fazenda	Patrão	600.000,00 Esc.	4	1	Camarote	Camarate
9	Fazenda	Patrão	600.000,00 Esc.	4	2	Informática	Covilhã
9	Fazenda	Patrão	600.000,00 Esc.	4	3	Produção	Guarda
9	Fazenda	Patrão	600.000,00 Esc.	4	4	Comercial	Lisboa
10	Marília	Secretária	85.000,00 Esc.	4	1	Camarote	Camarate
10	Marília	Secretária	85.000,00 Esc.	4	2	Informática	Covilhã
10	Marília	Secretária	85.000,00 Esc.	4	3	Produção	Guarda
10	Marília	Secretária	85.000,00 Esc.	4	4	Comercial	Lisboa

h) Junção (equi-join)

"Obtenha uma listagem com os empregados e os seus departamentos."

```
SELECT *  
FROM Empregado, Departamento  
WHERE Empregado.DepNum = Departamento.DepNum
```

Conjunto resultado:

Emp Num	Empregado. Nome	Categoria	Salario	Empregado. DepNum	Departamento. DepNum	Departamento. Nome	Local
1	José	Artista	230.000,00 Esc.	1	1	Camarote	Camarate
4	Zezinho	Artes gráficas	150.000,00 Esc.	1	1	Camarote	Camarate
2	Artur	Programador	250.000,00 Esc.	2	2	Informática	Covilhã
3	Carlos	Analista	300.000,00 Esc.	2	2	Informática	Covilhã
5	Abílio	Chefe secção	100.000,00 Esc.	3	3	Produção	Guarda
6	António	Design	100.000,00 Esc.	3	3	Produção	Guarda
7	Daniela	Modelo	80.000,00 Esc.	3	3	Produção	Guarda
8	Margarida	Director	350.000,00 Esc.	4	4	Comercial	Lisboa
9	Fazenda	Patrão	600.000,00 Esc.	4	4	Comercial	Lisboa
10	Marília	Secretária	85.000,00 Esc.	4	4	Comercial	Lisboa

i) Junção, Restrição e Projectão

"Obtenha uma listagem com os nomes dos empregados e dos respectivos departamentos."

```
SELECT E.Nome, D.Nome  
FROM Empregado E, Departamento D  
WHERE E.DepNum = D.DepNum  
And E.Categoria = 'Programador'
```

Conjunto resultado:

E. Nome	D. Nome
Artur	Informática

8. Mais experiências com SQL

Considere as ocorrências apresentadas no ponto anterior.

a) Contar ocorrências

"Quantos empregados trabalham no projecto 3?"

```
SELECT Count(*) AS Trabalhadores
FROM Atribuicao
WHERE ProjNum = 3
```

Conjunto resultado:

Trabalhadores
4

b) Somar valores

"Despesa mensal em salários com os trabalhadores do projecto 3?"

```
SELECT Sum(Salario) AS Total_Salarios
FROM Atribuicao A, Empregado E
WHERE ProjNum = 3
and A.EmpNum = E.EmpNum
```

Conjunto resultado:

Total_Salarios
580000

c) Valores Mínimo e Máximo

"Menor e maior salário entre os trabalhadores do projecto 3?"

```
SELECT Min(Salario) AS Menor_Salario,
Max(Salario) AS Maior_Salario
FROM Atribuicao A, Empregado E
WHERE ProjNum = 3
and A.EmpNum = E.EmpNum
```

Conjunto resultado:

Menor_Salario	Maior_Salario
80000	300000

d) Valor médio

"Valor médio do salário dos trabalhadores do projecto 3?"

```
SELECT Avg(Salario) AS Salario_Medio
FROM Atribuicao A, Empregado E
WHERE ProjNum = 3
and A.EmpNum = E.EmpNum
```

Conjunto resultado:

Salario_Medio
145000

9. Ainda mais experiências com SQL

Considere as ocorrências apresentadas.

Execute as interrogações das alíneas seguintes:

a) Contar ocorrências por grupos

"Quantos empregados trabalham em cada projecto?"

```
SELECT ProjNum, Count(*) AS Trabalhadores
FROM Atribuicao
GROUP BY ProjNum
```

Conjunto resultado:

ProjNum	Trabalhadores
1	1
2	2
3	4

b) Somar valores distribuídos por grupos

"Despesa mensal em salários com os trabalhadores de cada projecto?"

```
SELECT A.ProjNum, Sum(Salario) AS Total_Salarios
FROM Atribuicao A, Empregado E
WHERE A.EmpNum = E.EmpNum
GROUP BY ProjNum
```

Conjunto resultado:

ProjNum	Total_Salarios
1	230000
2	550000
3	580000

c) Somar valores distribuídos por grupos, impondo condições

" Projectos com despesa mensal superior a 500000\$"

```
SELECT A.ProjNum, Sum(Salario) AS Total_Salarios
FROM Atribuicao A, Empregado E
WHERE A.EmpNum = E.EmpNum
GROUP BY ProjNum
Having Sum(Salario) > 500000
```

Conjunto resultado:

ProjNum	Total_Salarios
2	550000
3	580000

d) Somar valores distribuídos por grupos, impondo condições e ordenação

" Projectos com despesa mensal superior a 500000\$ ordenado por ordem decrescente do valor da despesa"

```
SELECT A.ProjNum, Sum(Salario) AS Total_Salarios
FROM Atribuicao A, Empregado E
WHERE A.EmpNum = E.EmpNum
GROUP BY ProjNum
Having Sum(Salario) > 500000
Order By Sum(Salario) Desc
```

Conjunto resultado:

ProjNum	Total_Salarios
3	580000
2	550000

10. Inserir, alterar e eliminar tuplos

a) Inserir um registo

"Inserir o projecto 123, com 125000\$ de fundos e designação Novo projecto"

```
INSERT INTO Projecto ( ProjNum, Designacao, Fundos )  
Values (123, 'Novo projecto' 125000)
```

b) Inserir um registo usando as facilidades do "MS ACCESS"

"Inserir um novo projecto. Os elementos entre parêntesis recto são pedidos ao utilizador"

```
INSERT INTO Projecto ( ProjNum, Designacao, Fundos )  
SELECT [ProjNum], [Designacao], [Fundos];
```

c) Inserir vários registos

"Suponha que a tabela Projectos_Pendentes contém os projectos cujo orçamento foi recentemente aprovado. Pretende-se passar estes elementos para a tabela Projectos."

Projectos_Pendentes:

ProjNum	Designacao	Fundos
1	TotoGolo	5.060.000,00 Esc.
2	TotoLoto	4.000.000,00 Esc.

```
INSERT INTO Projecto ( ProjNum, Designacao, Fundos )  
SELECT ProjNum + 100, 'Novos:' + Designacao, Fundos  
FROM Projectos_Pendentes;
```

Projecto:

ProjNum	Designacao	Fundos
1	Rir à gargalhada	60.000,00 Esc.
2	BD do IPG	5.000.000,00 Esc.
3	Sapataria da Beira	3.000.000,00 Esc.
4	Lotaria	5.060.000,00 Esc.
101	Novos:TotoGolo	5.060.000,00 Esc.
102	Novos:TotoLoto	4.000.000,00 Esc.

d) Alterar um registo

"Alterar a designação do projecto 101 para TotoGolo e os fundos para 5000000\$"

```
UPDATE Projecto
SET   Designacao = 'TotoGolo',
      Fundos = 5000000
WHERE ProjNum = 101
```

Projecto:

ProjNum	Designacao	Fundos
1	Rir à gargalhada	60.000,00 Esc.
2	BD do IPG	5.000.000,00 Esc.
3	Sapataria da Beira	3.000.000,00 Esc.
4	Lotaria	5.060.000,00 Esc.
101	TotoGolo	5.000.000,00 Esc.
102	Novos:TotoLoto	4.000.000,00 Esc.

e) Apagar um registo

"Apagar os detalhes do projecto 101"

```
Delete From Projecto
WHERE ProjNum = 101
```

Projecto:

ProjNum	Designacao	Fundos
1	Rir à gargalhada	60.000,00 Esc.
2	BD do IPG	5.000.000,00 Esc.
3	Sapataria da Beira	3.000.000,00 Esc.
4	Lotaria	5.060.000,00 Esc.
102	Novos:TotoLoto	4.000.000,00 Esc.

f) Apagar vários registos

"Apagar os projectos cuja designação se inicia por Novos "

```
Delete From Projecto
WHERE Designacao Like 'Novos*'
```

Projecto:

ProjNum	Designacao	Fundos
1	Rir à gargalhada	60.000,00 Esc.
2	BD do IPG	5.000.000,00 Esc.
3	Sapataria da Beira	3.000.000,00 Esc.
4	Lotaria	5.060.000,00 Esc.

11. Formulário inicial

Por vezes, quando se inicia uma aplicação, pretende-se ter um formulário (formulário inicial ou de *startup*) que dá uma visão rápida acerca do que trata a aplicação.

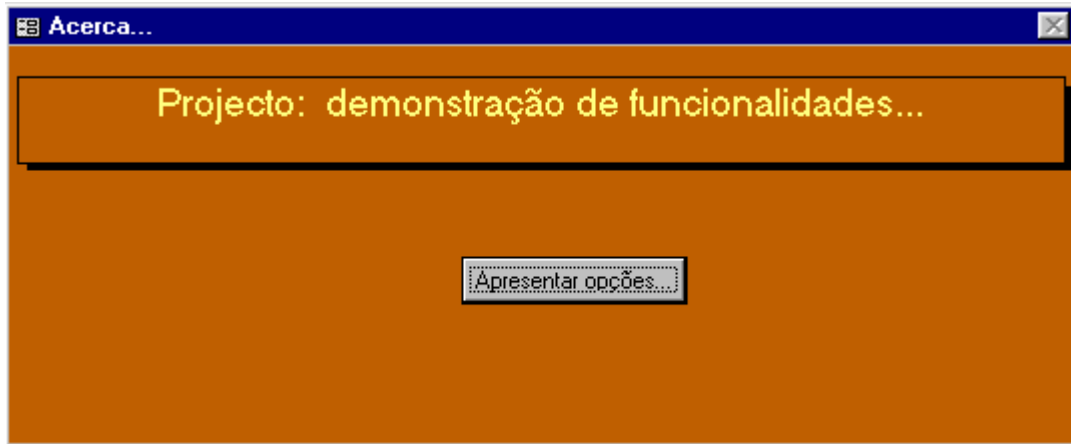


Fig. 9 Formulário inicial.

O formulário inicial pode ser construído usando o assistente de formulários (activado através da selecção do botão *Novo*, do separador formulários, e depois escolhendo a opção *Vista de estrutura* no diálogo obtido).

No formulário inicial podem ser colocados diversos elementos, por exemplo, um título para a aplicação e um botão para iniciar os trabalhos. Naturalmente, para que o botão iniciar possa activar o desenrolar do trabalho, deve ter associado algum código de activação (escrito em *Visual Basic*).

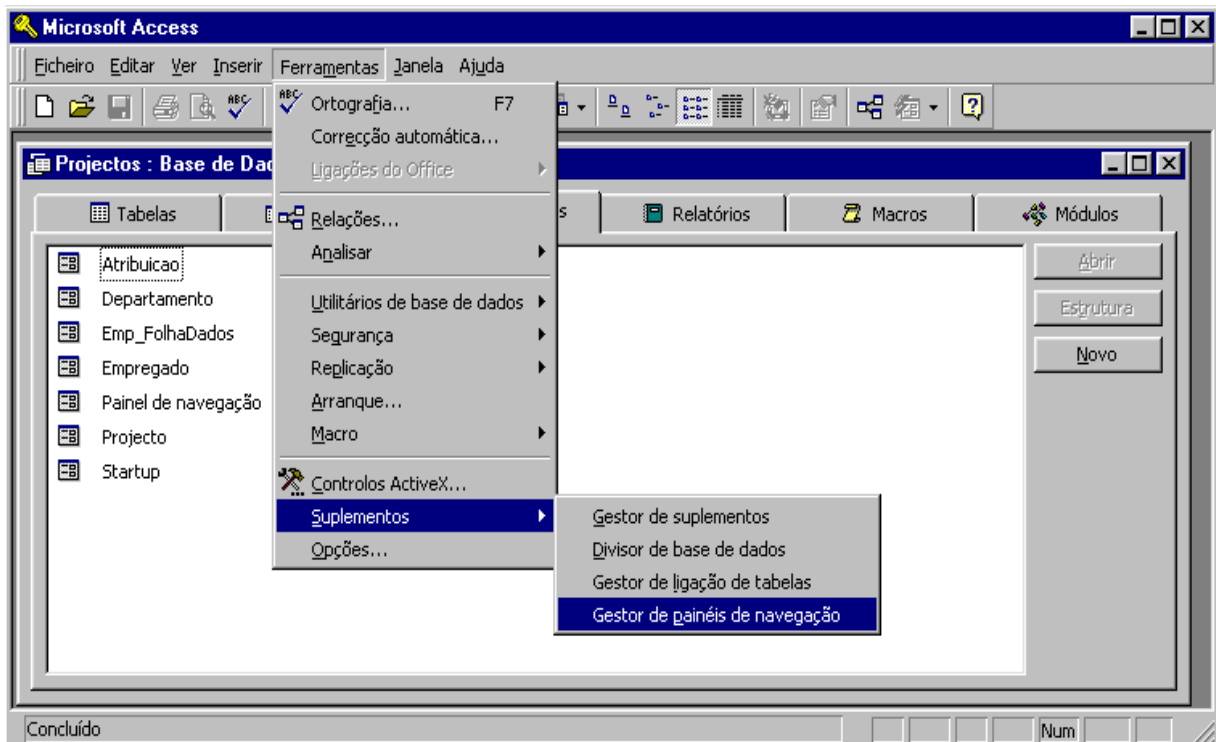
Nota: Explore o formulário *Startup* e o código do módulo *Startup* contido no separador Módulos.

Portanto, o formulário inicial funciona como uma porta de entrada para a aplicação.

12. Painéis de navegação

O MS Access dispõe de painéis de navegação que podem ser usados na construção de aplicações.

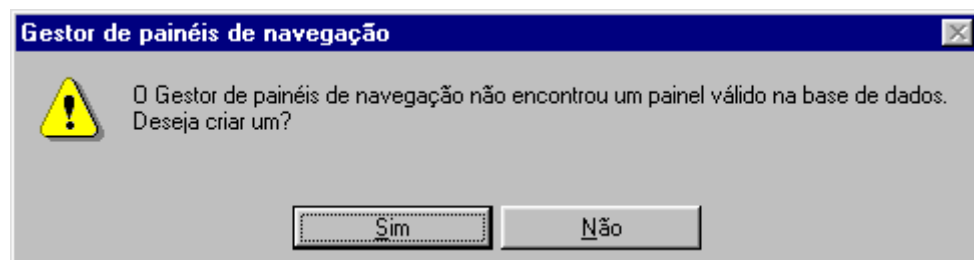
Para activar o gestor de painéis de navegação deve seleccionar o submenu *Suplementos* do menu *Ferramentas* e escolher a opção *Gestor de painéis de navegação*.



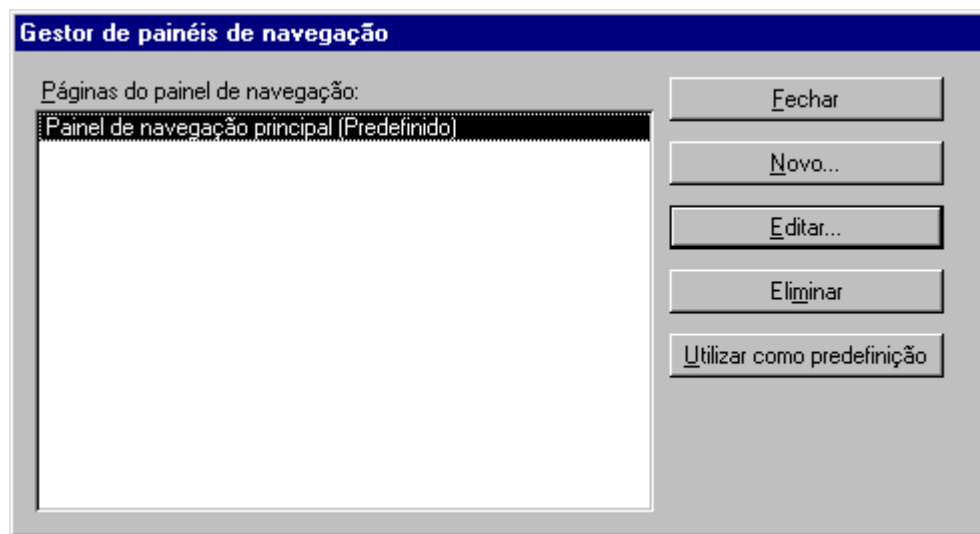
Usando o gestor de painéis de navegação deve criar os painéis de navegação necessários para o desenrolar da aplicação. Um painel sempre útil é o painel principal que mostra as opções de nível global.

Para criar o painel principal proceda da forma seguinte:

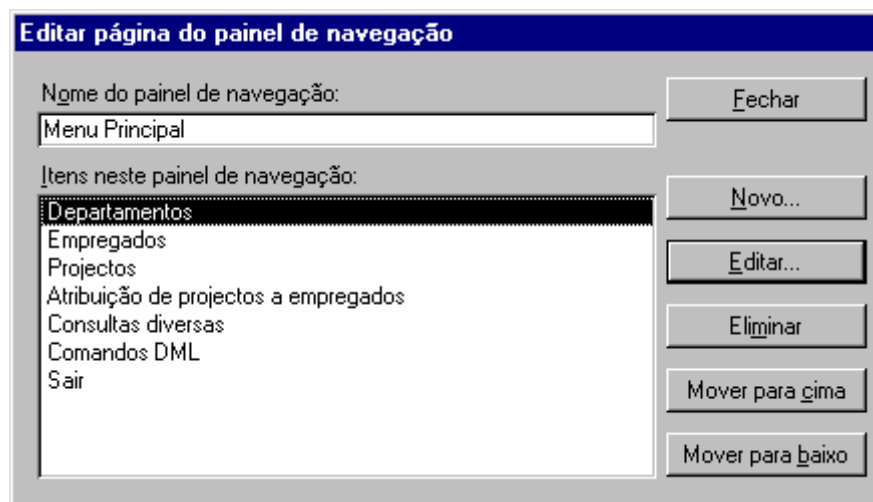
→ Active o gestor de painéis de navegação e informe o gestor para criar um painel.



→ Edite o painel de navegação principal.

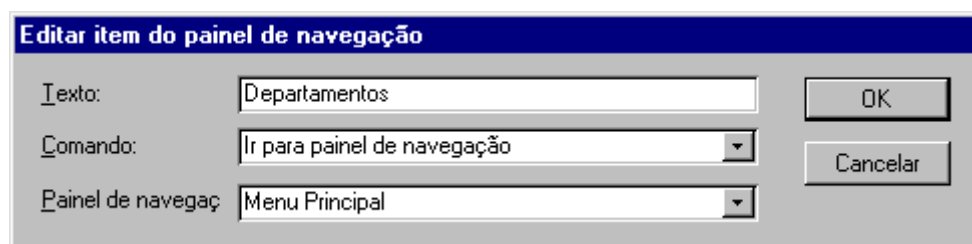


→ Escolha um nome para o painel e inclua os itens a mostrar neste painel.

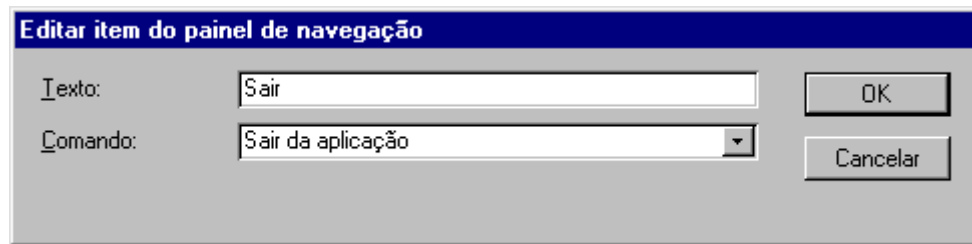


→ Para cada item criado escolha as opções apropriadas.

Inicialmente escolha as opções da forma apresentada na figura:



Inclua uma opção para sair da aplicação:



Editar item do painel de navegação

Texto: Sair

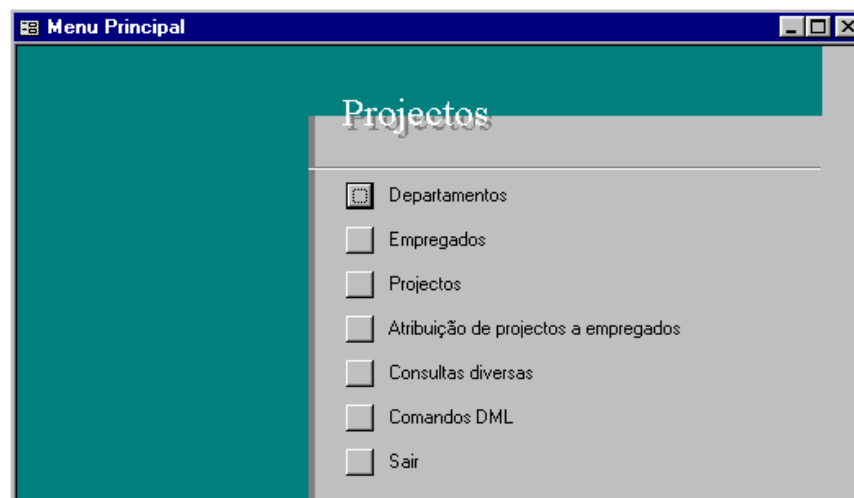
Comando: Sair da aplicação

OK

Cancelar

→ Conclua a edição do painel de navegação e teste a aplicação.

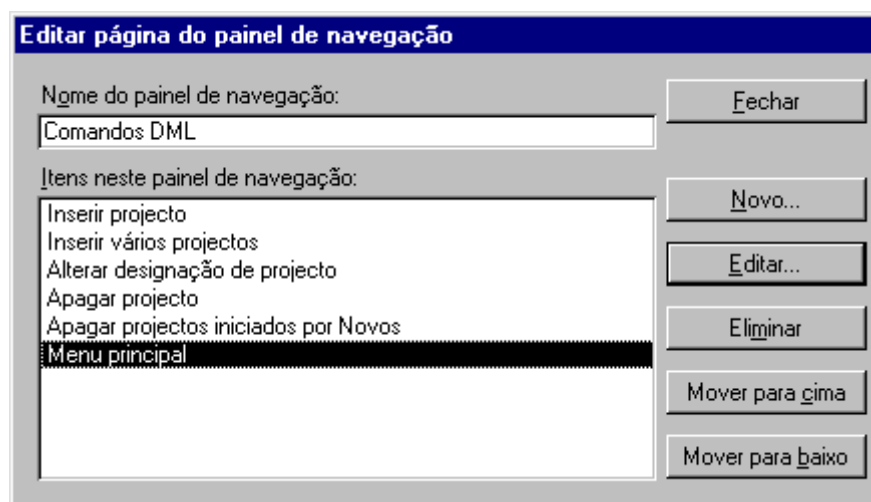
Obtém-se um resultado semelhante ao seguinte:



Para criar outros painéis:

→ Active o gestor de painéis de navegação, crie um novo painel e dê-lhe o nome de Comandos DML.

→ Acrescente alguns itens ao novo painel:



Editar página do painel de navegação

Nome do painel de navegação: Comandos DML

Itens neste painel de navegação:

- Inserir projecto
- Inserir vários projectos
- Alterar designação de projecto
- Apagar projecto
- Apagar projectos iniciados por Novos
- Menu principal

Fechar

Novo...

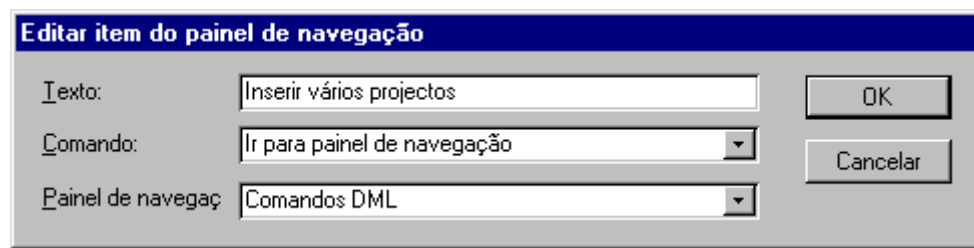
Editar...

Eliminar

Mover para cima

Mover para baixo

→ Para cada item estabeleça a acção apropriada, por exemplo:



Editar item do painel de navegação

Texto:

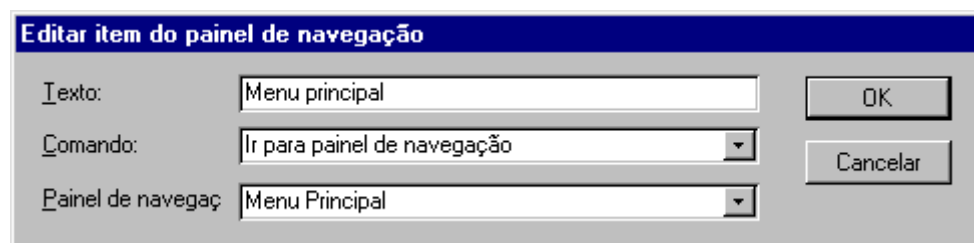
Comando:

Painel de navegação:

OK

Cancelar

→ Inclua uma opção para voltar ao menu principal:



Editar item do painel de navegação

Texto:

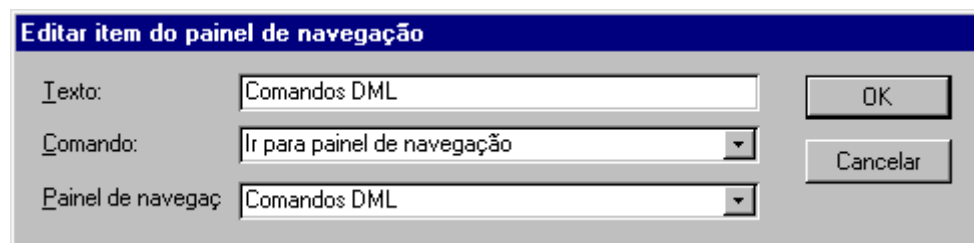
Comando:

Painel de navegação:

OK

Cancelar

→ Termine a edição do painel corrente e edite o painel principal. Edite o item Comandos DML e estabeleça o painel de navegação apropriado:



Editar item do painel de navegação

Texto:

Comando:

Painel de navegação:

OK

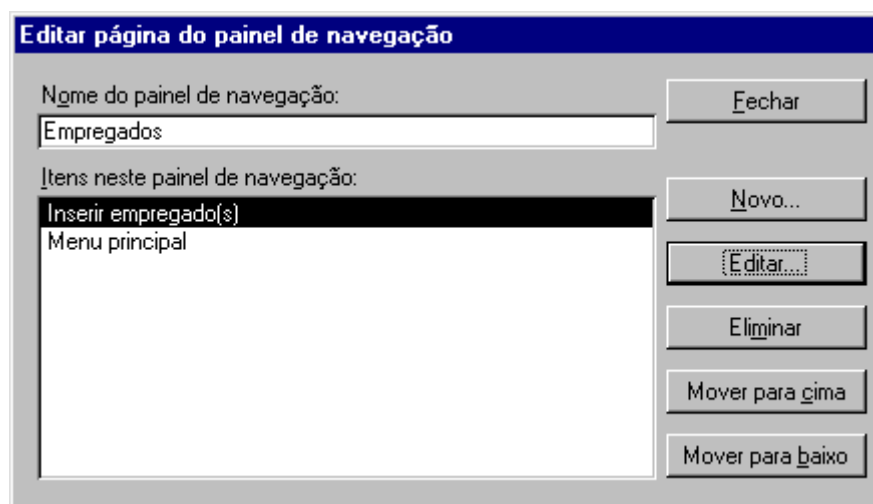
Cancelar

→ Feche o gestor e teste a aplicação.

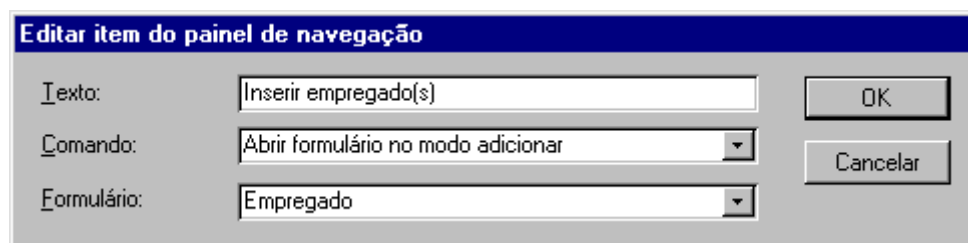
Para criar painéis para as outras opções do painel principal:

Proceda de forma semelhante para os outros itens do painel principal, i.e., crie painéis de navegação para cada opção.

→ Exemplo: painel de navegação Empregados em edição.



→ Exemplo: opção para inserir um empregado.



Notar que o *comando* indica a acção *Abrir formulário no modo adicionar* e na opção formulário está especificado o formulário a usar.