

Conectando Bancos de Dados Microsoft Access no BrOffice.org Base

fornecido pelo

Projeto de Documentação do BrOffice.org

Índice

| | | |
|-----|---|----|
| 1 | Introdução..... | 2 |
| 1.1 | Versão..... | 2 |
| 1.2 | Licenciamento..... | 2 |
| 1.3 | Mensagem do Projeto BrOffice.org..... | 2 |
| 2 | Conectando Bancos de Dados Access..... | 3 |
| 2.1 | Tabelas..... | 5 |
| 2.2 | Exibições..... | 6 |
| 2.3 | Criando Consultas..... | 9 |
| 2.4 | Diferenças entre Consultas e Exibições..... | 20 |
| 2.5 | Relacionamentos..... | 21 |
| 3 | Créditos e Agradecimentos..... | 25 |
| 3.1 | Créditos..... | 25 |
| 3.2 | A Fazer | 25 |
| 3.3 | Agradecimentos..... | 25 |
| 3.4 | Bibliografia | 25 |
| 3.5 | Histórico do Documento..... | 25 |

1 Introdução

1.1 Versão

Conectando Bancos de Dados Microsoft Access no BrOffice.org Base, versão 0.9.

A versão em português do Brasil deste documento foi desenvolvida a partir do modelo de documentos do BrOffice.org, disponível em <http://www.broffice.org>.

1.2 Licenciamento

Este trabalho está licenciado sob uma Licença Creative Commons Atribuição-UsoNãoComercial-Compartilhamento pela mesma licença. Para ver uma cópia desta licença, visite <http://creativecommons.org/licenses/by-nc-sa/2.0/br/> ou envie uma carta para Creative Commons, 559 Nathan Abbott Way, Stanford, California 94305, USA.

1.3 Mensagem do Projeto BrOffice.org

Periodicamente, o Projeto BrOffice.org fará a liberação da versão oficial deste documento, atualizando, com certificação técnica, quaisquer conteúdos que venham a ser incluídos no texto. Colaborações podem ser enviadas para o e-mail gbpacheco@BrOffice.org.

O Projeto BrOffice.org, ou qualquer outro colaborador, não se responsabiliza por qualquer dano direto ou indireto causado pela utilização deste documento, cabendo ao próprio usuário a responsabilidade sobre qualquer problema que possa vir a ocorrer.

A última versão atualizada deste documento poderá ser encontrada em: <http://www.broffice.org.br>.

Solicitamos e recomendamos que o conteúdo desta obra e suas eventuais derivações sejam armazenadas em formatos de arquivos livres e padronizados e não em formatos proprietários e fechados.

Saudações e bom trabalho!

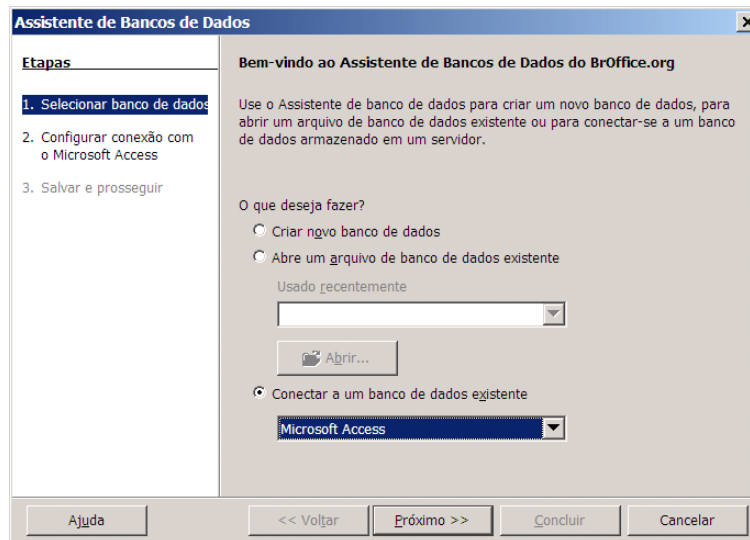
A Coordenação do Projeto BrOffice.org

2 Conectando Bancos de Dados Access

O primeiro passo para utilizar um banco de dados do Access no BrOffice.org é verificar a estrutura do arquivo .mdb correspondente. Ou seja, antes de mais nada, é necessário certificar que a forma como as informações estão armazenadas é compatível com as funcionalidades do BrOffice.org.

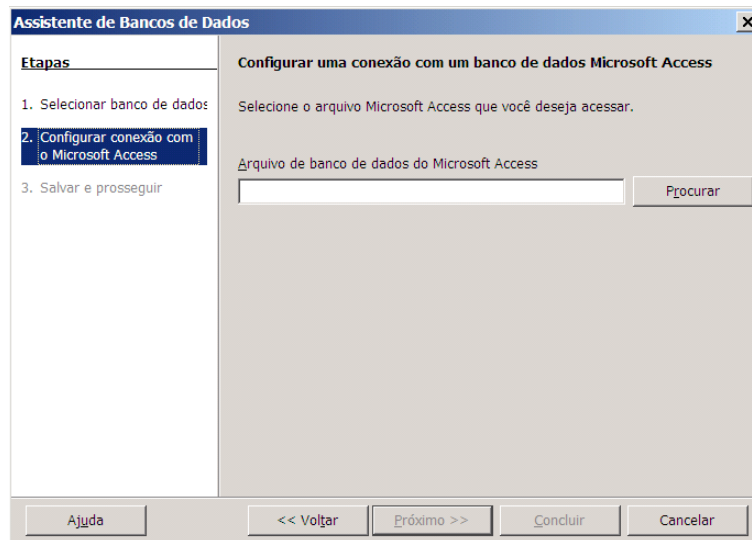
Ao abrir o arquivo .mdb, ainda no Microsoft Access, certifique-se de que os nomes das estruturas do banco (como tabelas, consultas, campos, etc) foi atribuído com o cuidado de não conter espaços e não conter caracteres especiais. Espaços, por exemplos podem ser substituídos pelo símbolo “_”.

Diferentemente do que foi apresentado na seção 1, agora, ao criarmos o nosso novo banco de dados em **Arquivo > Novo > Banco de Dados**, vamos escolher a opção **Conectar a um banco de dados existente**, que nada mais é do que vincularmos a interface do BrOffice.org ao arquivo de banco de dados .mdb. Para isso, escolheremos na lista imediatamente abaixo, o tipo **Microsoft Access**.

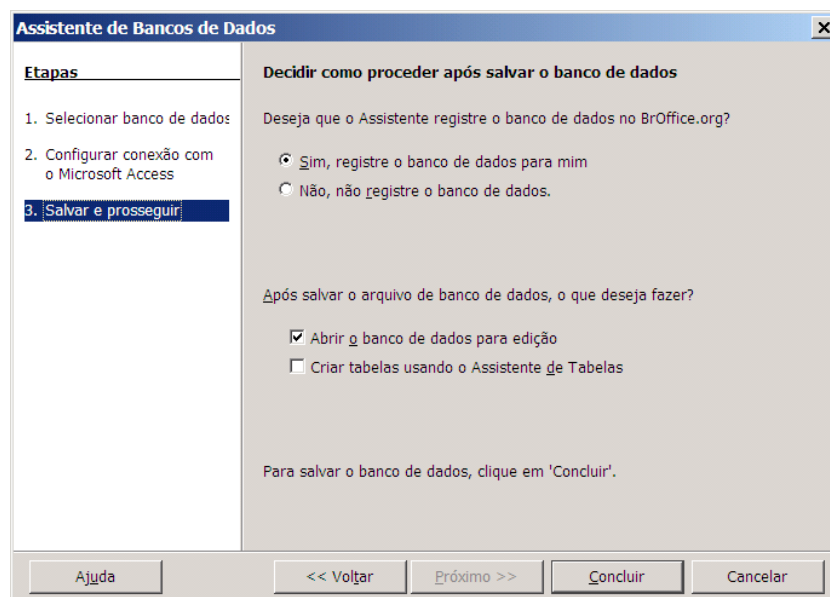


No próximo passo, indique o arquivo .mdb que deverá ser aberto pelo BrOffice.org.

Para indicar o caminho e o nome do arquivo, use o botão **Procurar**.

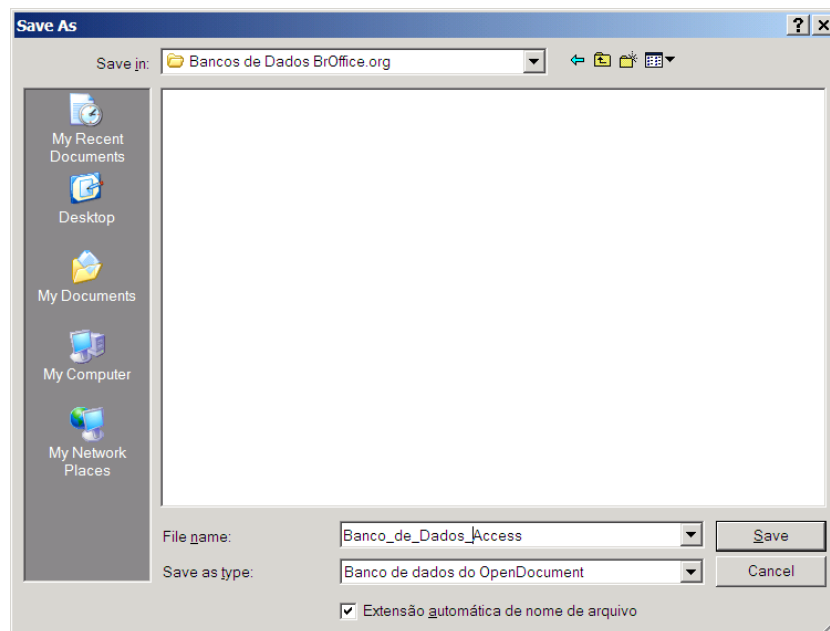


Na próxima tela possui a definição das mesmas funcionalidades especificadas na seção 1: registrar ou não o banco de dados para uso das demais aplicações do BrOffice.org, abrir ou não o banco de dados para edição e criar ou não novas tabelas usando o Assistente de Tabelas.

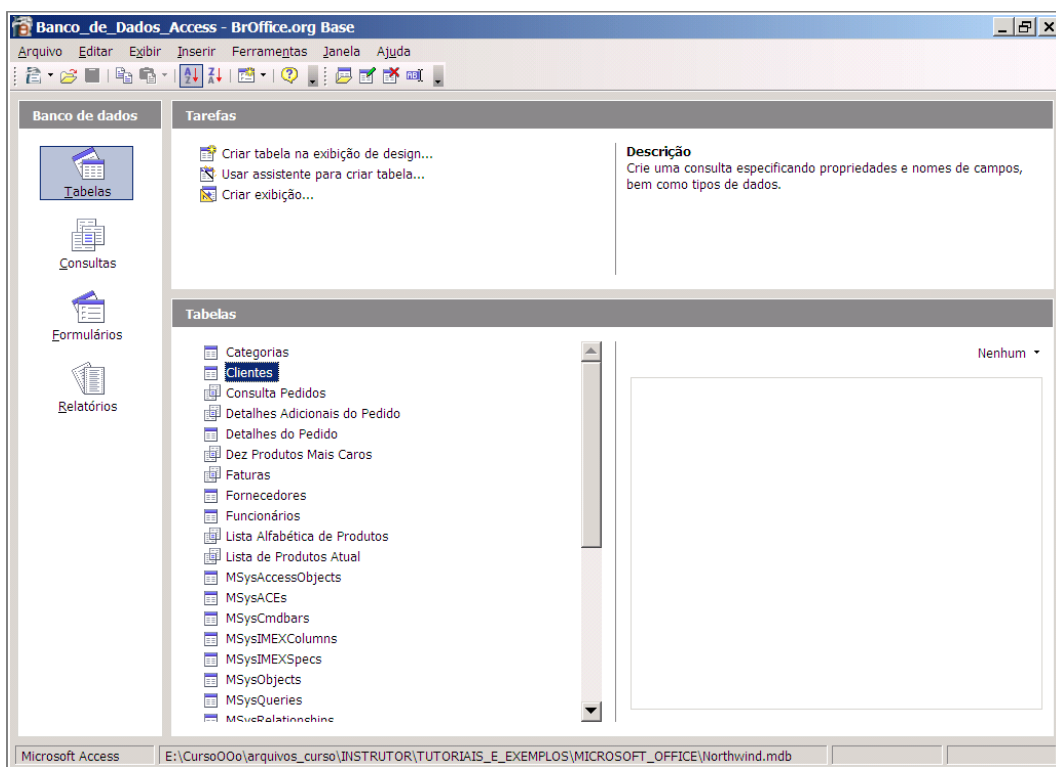


Mantenha definidas as propriedades padrões e clique em **Concluir**.

Será aberta a tela de salvamento do arquivo **.odb**. Esse arquivo conterá as definições do vínculo entre o BrOffice.org e o arquivo **.mdb**.



Depois de criar o arquivo, o BrOffice.org abrirá o mesmo para edição, apresentando todas as tabelas e consultas do Access. Formulários e relatórios do Access não são apresentados no BrOffice.org.



2.1 Tabelas

Tabelas são estruturas de armazenamento de dados. Os dados são armazenados em registros. Já vimos, na seção 2, como fazemos para criar uma tabela. Agora, aprofundaremos um pouco mais o assunto, diferenciando dois tipos de estruturas que podem ser encontradas dentro desta seção do Banco de Dados. Quando utilizamos um arquivo .mdb do Access como fonte de dados, podemos verificar que as tabelas inseridas nessa categoria são as tabelas e as consultas do banco de dados .mdb. Através dos ícones, é possível identificar a origem da estrutura:



Tabela



Consulta do Access ou Exibição do BrOffice.org

Há uma diferença significativa entre as duas estruturas. A tabela, como já vimos, é uma estrutura que permite a manipulação completa dos registros armazenados (visualização, inclusão, modificação e exclusão).

Já a Exibição é significativamente diferente, pois é apenas a visualização de um conjunto de vários campos de várias tabelas em uma única estrutura, similar a uma tabela. É um recurso que serve, por exemplo, de base para a criação de relatórios.

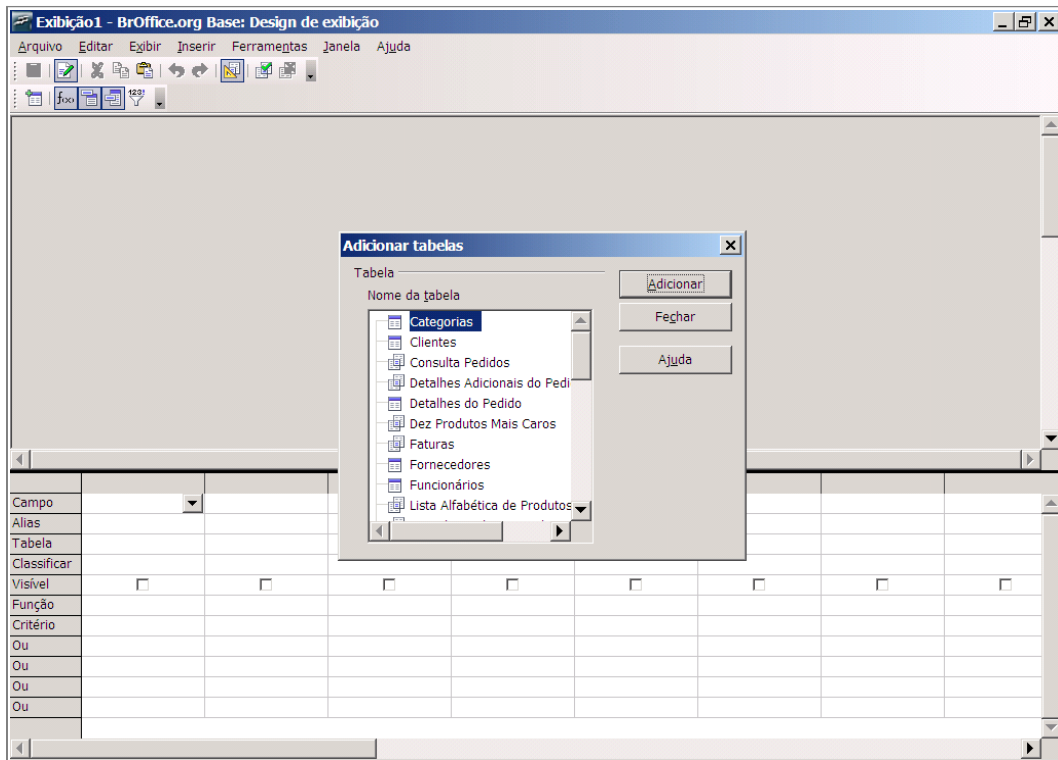
As consultas do Access são apresentadas, no BrOffice.org, como exibições. Isso significa que uma consulta do Access não será alterada no BrOffice.org caso tenha sido incluída no arquivo .mdb.

2.2 Exibições

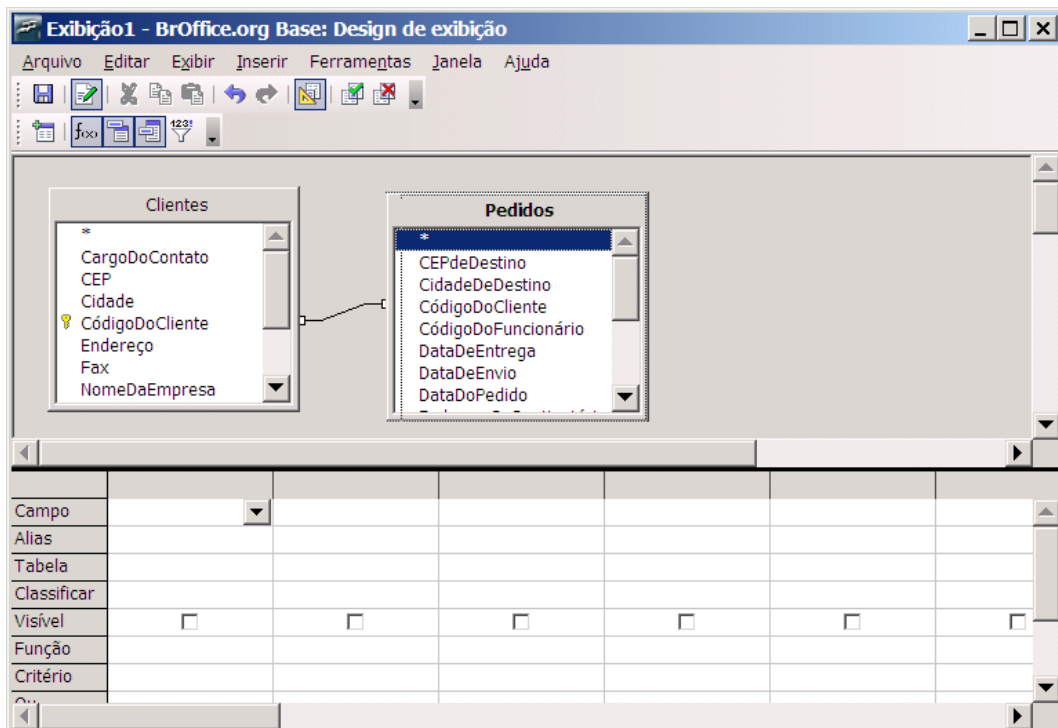
Para criar uma exibição, selecione a seção Tabelas e clique em **Criar exibição...** na área de Tarefas.



A tela de definição da exibição é muito similar à tela de criação de uma consulta. Podemos escolher as tabelas no diálogo Adicionar tabelas. Ao adicionar uma tabela, automaticamente são apresentados os relacionamentos entre a tabela incluída e as tabelas já inseridas. A partir das tabelas exibidas, escolheremos os campos que farão parte da exibição.



Na tela abaixo, fizemos a inclusão das tabelas Clientes e Pedidos.




Podemos, então, escolher os campos que farão parte da exibição. Podemos, por exemplo, definir que serão exibidos todos os pedidos por cliente.

Para isso, escolheremos os campos adequados de cada uma das tabelas.

| | | | | |
|-------------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| Campo | NomeDaEmpresa | NomeDoContato | Telefone | NúmeroDoPedido |
| Alias | | | | |
| Tabela | | Clientes | | |
| Classificar | | | | |
| Visível | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Função | | | | |
| Critério | | | | |

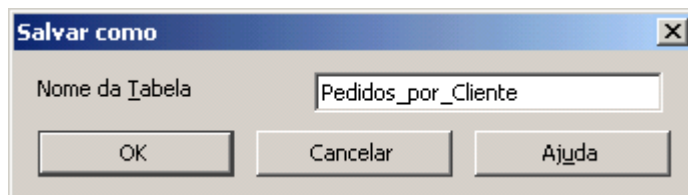
Vamos exibir, portanto, o nome da empresa, o nome do contato, o seu telefone e os números de pedidos correspondentes. Note que poderíamos, se fosse desejado, utilizar as funções para contabilizarmos algum resultado específico.

| | | | | |
|----------|-------------------------------------|-------------------------------------|-------------------------------------|-------------------------------------|
| Visível | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| Função | | | | sem função |
| Critério | | | | Média |
| Ou | | | | Contagem |
| Ou | | | | Máximo |
| Ou | | | | Mínimo |
| ~ | | | | Soma |

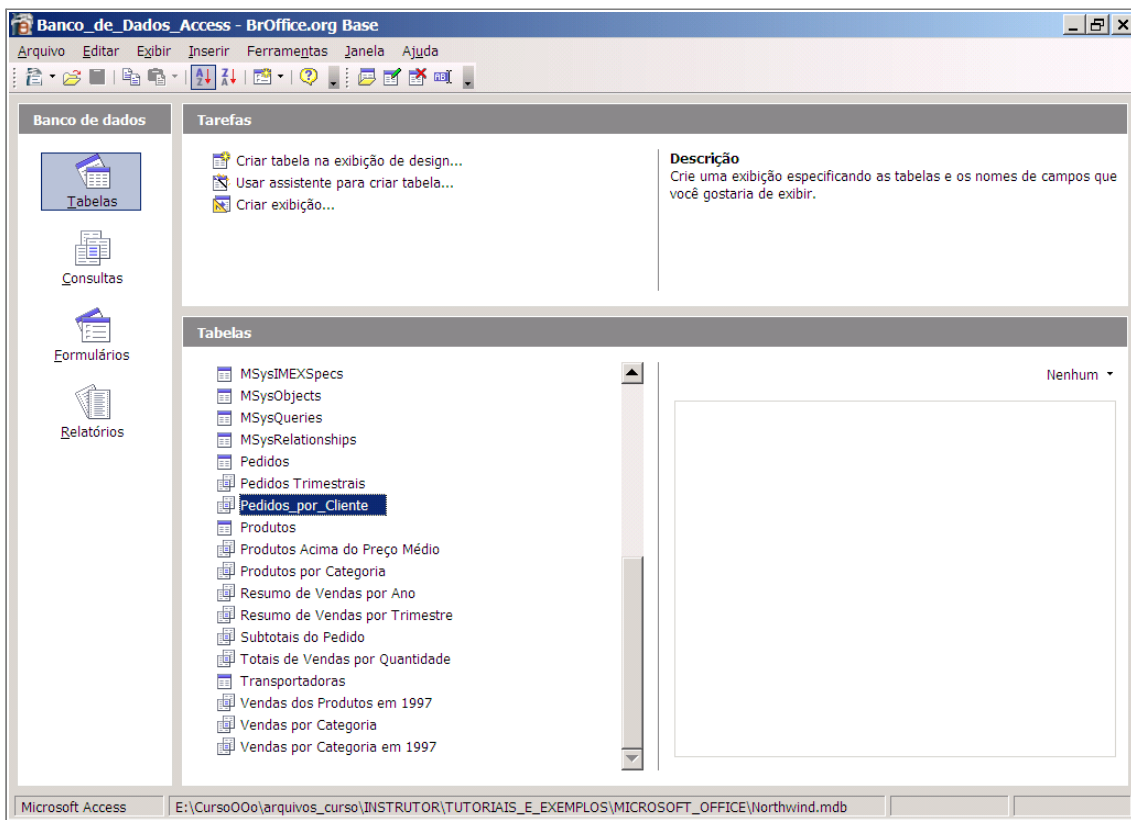
Ao final da definição da Exibição, vamos salvá-la clicando no botão Salvar . Será aberta a tela **Salvar como**, onde indicaremos um nome significativo para a exibição.



Lembramos que sempre é recomendável que os nomes das tabelas, campos, consultas, exibições e todas as outras estruturas do banco de dados sejam salvas sem acentos, espaços ou caracteres especiais. Por causa disso, daremos o nome a nossa exibição de Pedidos_por_Cliente.




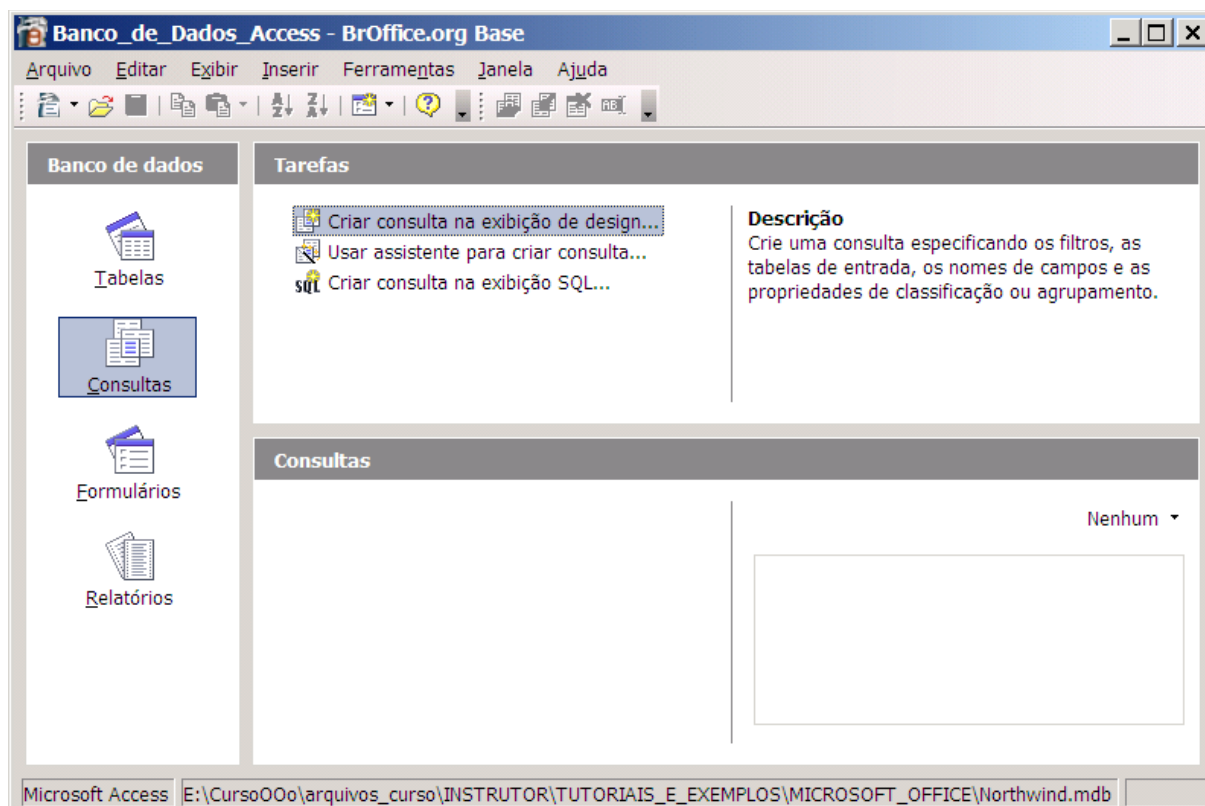
Ao finalizar o salvamento, a exibição será apresentada juntamente com as demais tabelas, com o seu ícone respectivo.



2.3 Criando Consultas

Na seção Consultas, poderão ser incluídas consultas SQL propriamente ditas. No entanto, as consultas incluídas aqui não serão inseridas no banco de dados .mdb.

 Consulta do BrOffice.org

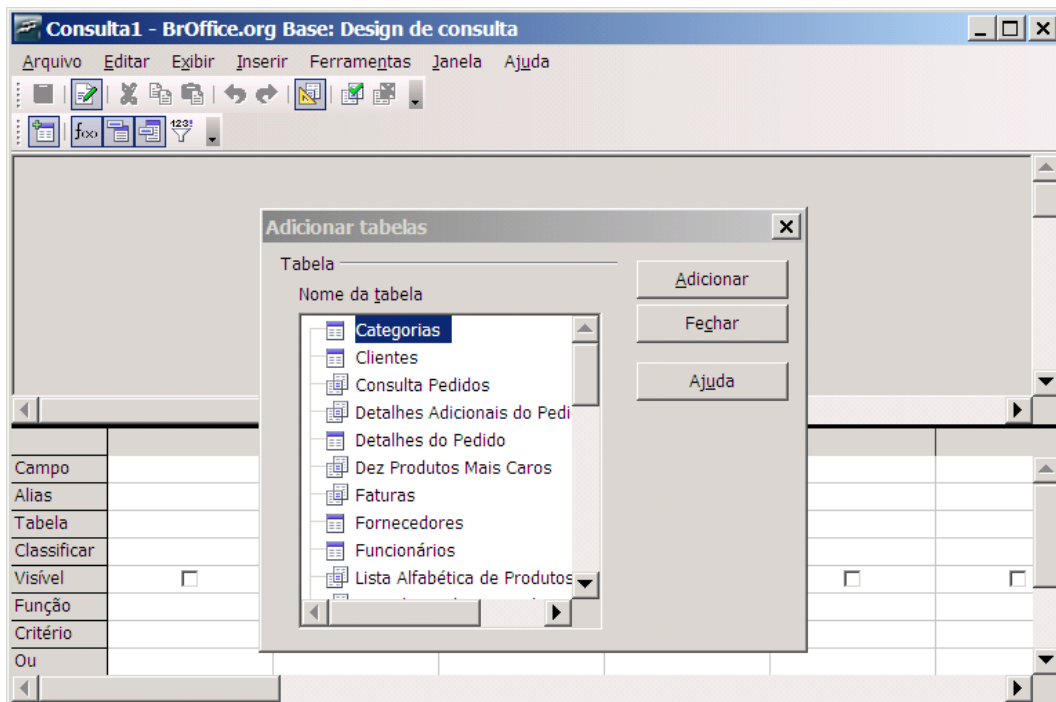


Note que, a primeira vez que clicamos na seção Consultas, não existe nenhuma consulta pré-existente. Podemos, então, criar uma consulta de três formas diferentes:

| Método | Descrição |
|--|--|
| Criar consulta na exibição de design... | É uma forma bastante tradicional de criação de consultas, muito similar ao método que é encontrado também no Microsoft Access. Através da escolha das tabelas e dos campos, o usuário cria visualmente a consulta. |
| Usar o assistente para criar consulta... | A criação da consulta é feita passo-a-passo, através de um assistente. É um recurso bastante útil para usuários que ainda não são familiarizados com as notações de bancos de dados. Importante: para a utilização do assistente é fundamental a correta instalação do JRE (Java Runtime Environment). |
| Criar consulta na exibição SQL... | Este é um método bastante interessante para os usuários de nível avançado que dominam a linguagem SQL. Basta inserir a consulta em SQL e executá-la que os resultados aparecerão na metade superior da tela. |

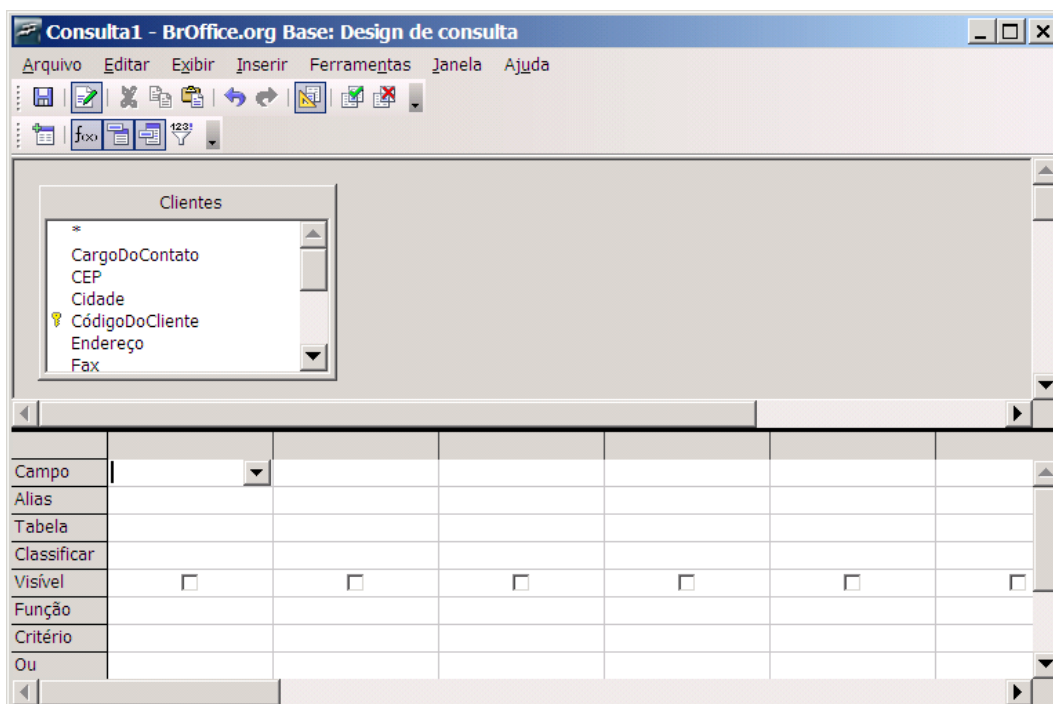
Criar consulta na exibição de design...

Ao selecionar a opção Criar consulta na exibição de design... será aberta a tela de criação da consulta através da escolha visual de tabelas, campos e opções.

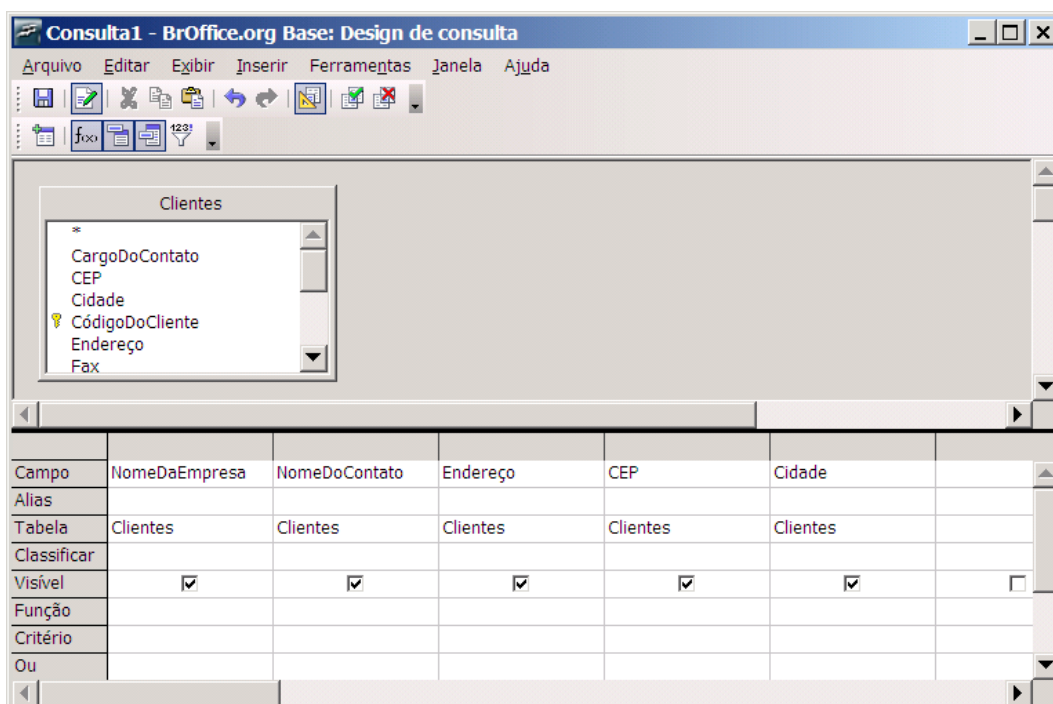


Criaremos, por exemplo, uma consulta que nos retornará dados dos clientes da cidade de Lisboa.

O primeiro passo é incluirmos a tabela de Clientes na consulta.




Depois disso, devemos escolher os campos que serão mostrados como resultado da consulta. Para o nosso exemplo, vamos escolher um conjunto básico de informações, como o nome da empresa, o nome do contato e os dados de localização. Os campos são escolhidos na área inferior da tela, na seqüência em que serão exibidos:

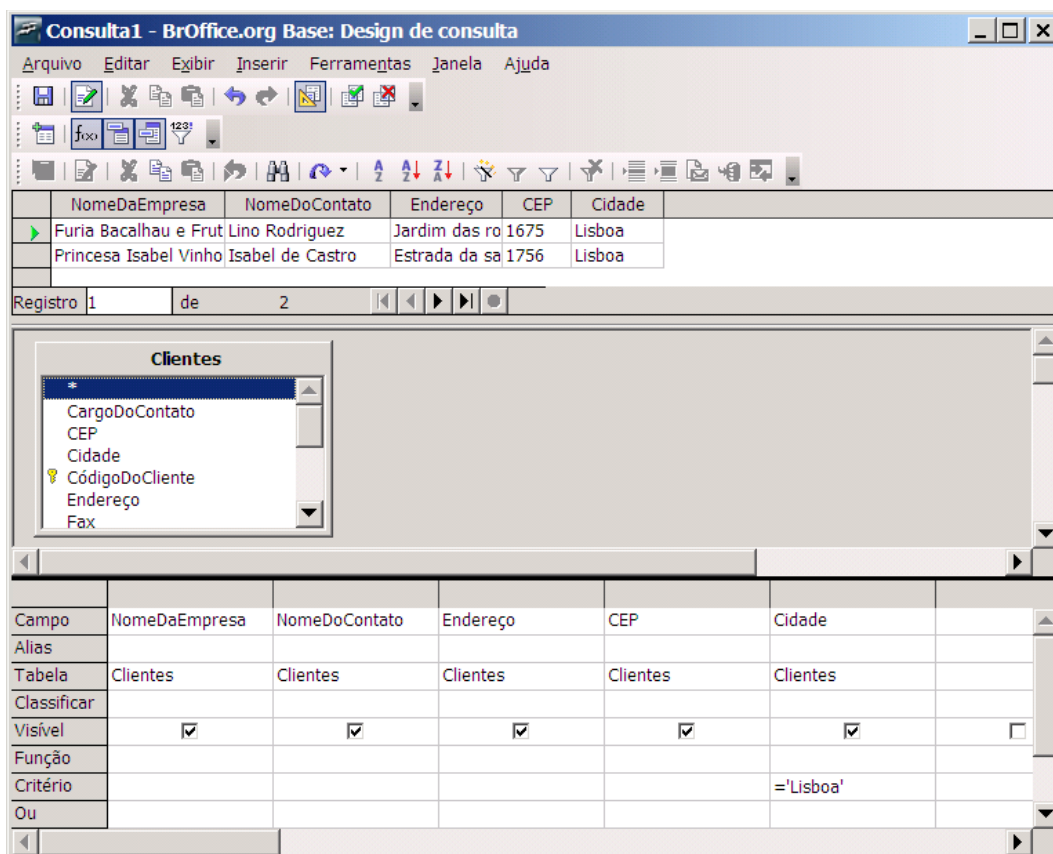


Note que ainda não estabelecemos nenhum critério extra para os campos, simplesmente escolhemos os campos que serão exibidos na consulta. No entanto, temos um critério estabelecido para o campo cidade, que é a restrição para os valores iguais a “Lisboa”.


Para inserirmos esse critério, podemos usar o campo “cidade” já incluído e complementarmos com as definições necessárias. Basta colocarmos, no campo critério, o sinal de igualdade mais o valor Lisboa entre aspas simples, como apresentado na figura abaixo.

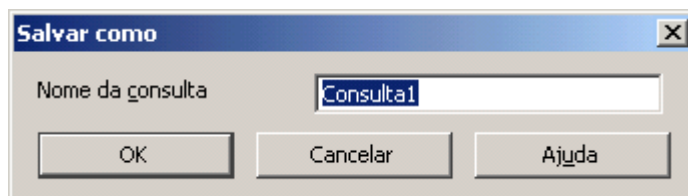
| | | | |
|-----|-------------------------------------|-------------------------------------|-------------------------------------|
| ato | Endereço | CEP | Cidade |
| | Clientes | Clientes | Clientes |
| | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| | | | = 'Lisboa' |

Finalizada a construção da consulta, o resultado pode ser exibido com o clique no botão , chamado Executar Consulta. O resultado é exibido acima da área de edição:



Com o critério estabelecido, foram retornados dois registros do banco de dados.

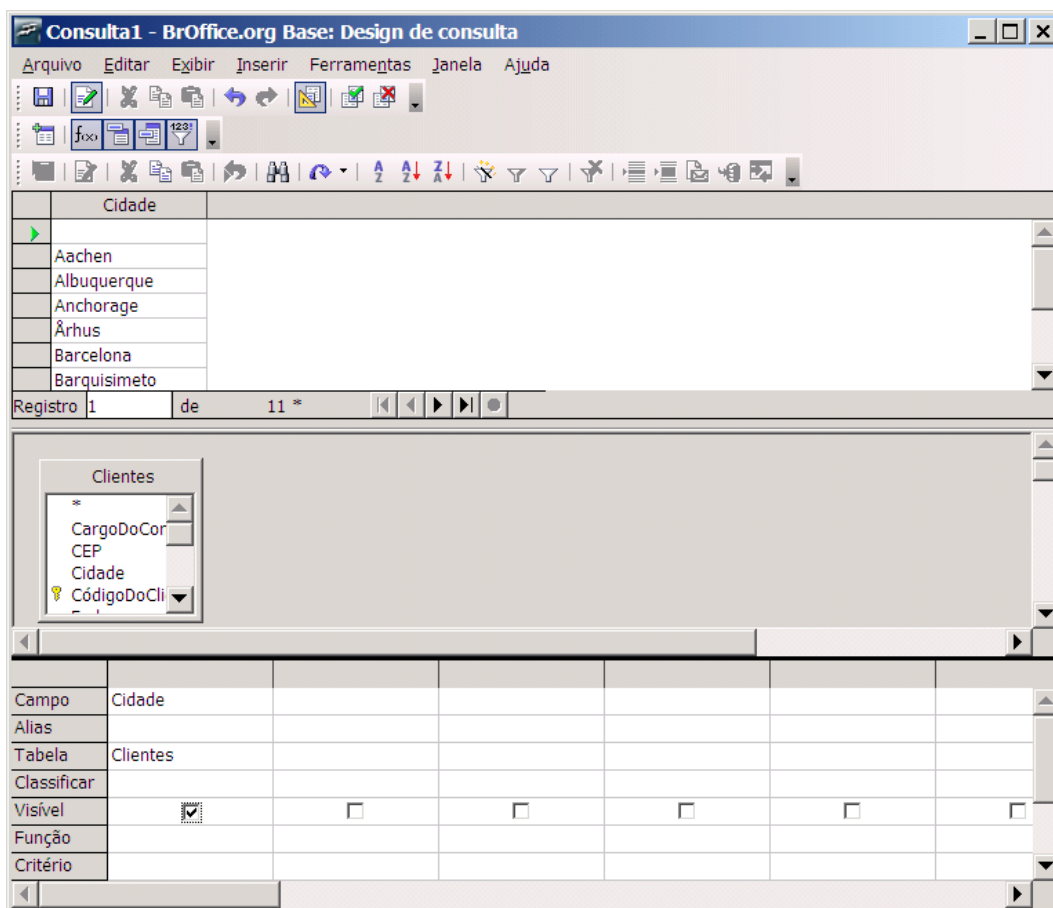
Por fim, podemos salvar a consulta clicando no botão Salvar .




No nome da consulta, colocaremos **Clientes_de_Lisboa** seguindo os mesmos critérios de nomenclatura sugeridos anteriormente.

Outros exemplos de consultas

1) Lista das cidades onde a empresa possui clientes.

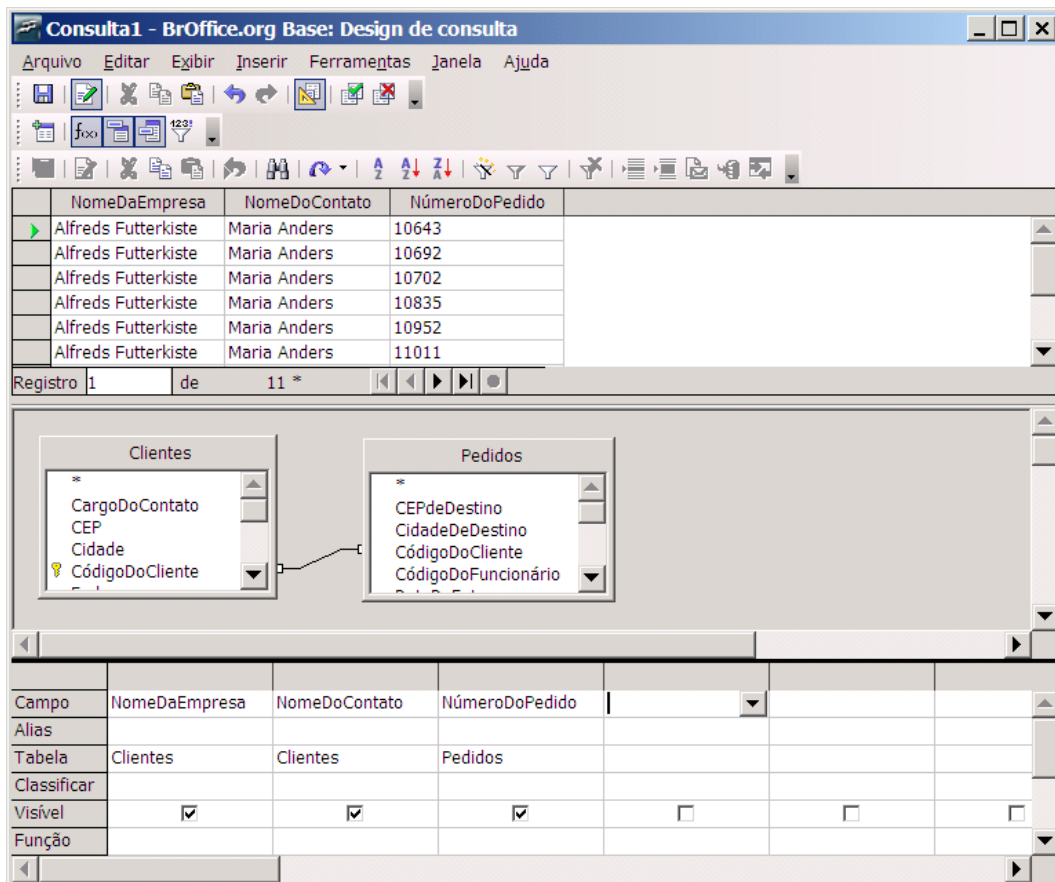


O exemplo da lista de cidades é muito simples, mas serve para identificarmos uma função bastante interessante. Ao selecionarmos apenas o campo Clientes.Cidade, retornaremos todas as cidades do banco de dados, repetindo o nome das cidades tantas vezes quanto for o número de clientes lá estabelecidos. Esse é um possível resultado, mas não é nada elegante mantê-lo assim. O ideal, nesse caso, é retornarmos como resultado o nome das cidades apenas uma vez para cada cidade. Isso pode ser feito com

a habilitação do botão  (Valores distintos). Esse botão faz com que as ocorrências de registros repetidos sejam representadas uma única vez no resultado final.

2) Informações de Clientes e Pedidos

O exemplo de informações de Clientes e Pedidos ilustra como podemos criar consultas com múltiplas tabelas. Incluímos as tabelas de Clientes e Pedidos e geramos um resultado que apresenta o nome da empresa e o nome do contato (informações da tabela Cliente) e o número do pedido correspondente (informação da tabela Pedido). Automaticamente, no momento da inclusão das tabelas, os relacionamentos são identificados através das chaves.



The screenshot shows the 'Design de consulta' window in BrOffice.org Base. The top part displays a query result with the following data:

| NomeDaEmpresa | NomeDoContato | NúmeroDoPedido |
|---------------------|---------------|----------------|
| Alfreds Futterkiste | Maria Anders | 10643 |
| Alfreds Futterkiste | Maria Anders | 10692 |
| Alfreds Futterkiste | Maria Anders | 10702 |
| Alfreds Futterkiste | Maria Anders | 10835 |
| Alfreds Futterkiste | Maria Anders | 10952 |
| Alfreds Futterkiste | Maria Anders | 11011 |

Below the data, a relationship diagram shows two tables: 'Clientes' and 'Pedidos'. The 'Clientes' table has fields: CargoDoContato, CEP, Cidade, and CódigoDoCliente. The 'Pedidos' table has fields: CEPdeDestino, CidadeDeDestino, CódigoDoCliente, and CódigoDoFuncionário. A line connects the 'CódigoDoCliente' field in 'Clientes' to the 'CódigoDoCliente' field in 'Pedidos', indicating a relationship.

At the bottom, a table structure view shows the following fields and their sources:

| Campo | NomeDaEmpresa | NomeDoContato | NúmeroDoPedido | | | |
|-------------|-------------------------------------|-------------------------------------|-------------------------------------|--------------------------|--------------------------|--------------------------|
| Alias | | | | | | |
| Tabela | Clientes | Clientes | Pedidos | | | |
| Classificar | | | | | | |
| Visível | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Função | | | | | | |

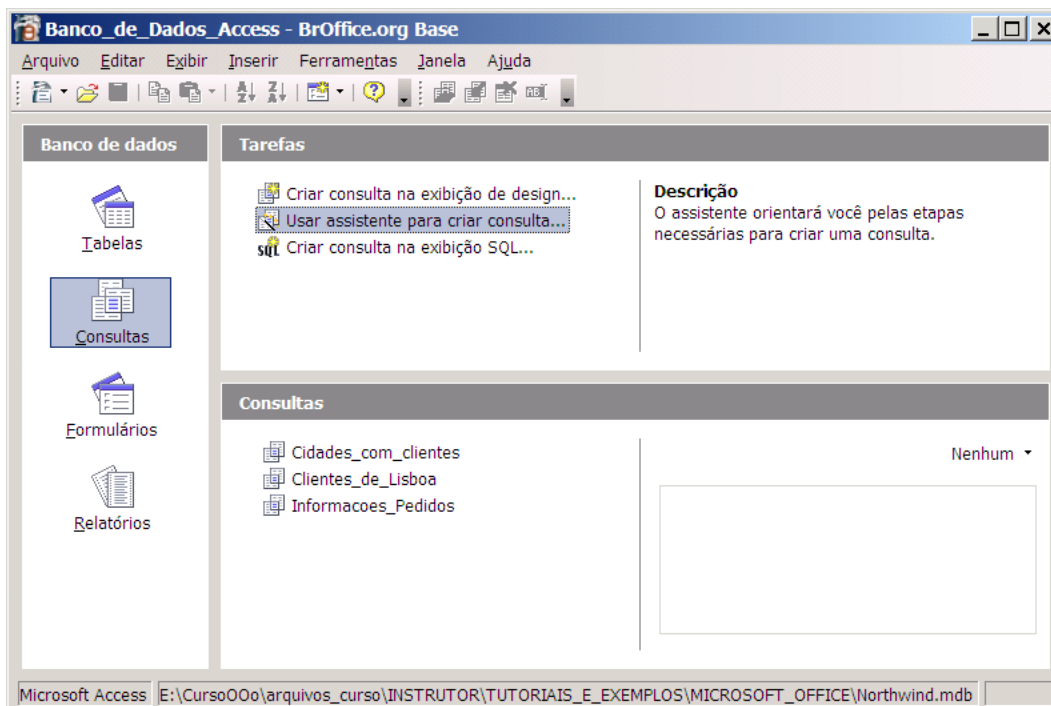
Para uma boa prática de criação de consultas é fundamental o conhecimento das estruturas das tabelas envolvidas e das relações criadas entre as mesmas.

Usando o assistente para criar consulta...

O assistente de criação de consultas é um instrumento prático para o usuário que não possui grande experiência na manipulação de bancos de dados. como pré-requisito para essa funcionalidade, existe a necessidade da instalação de uma JRE (Java Runtime Environment).

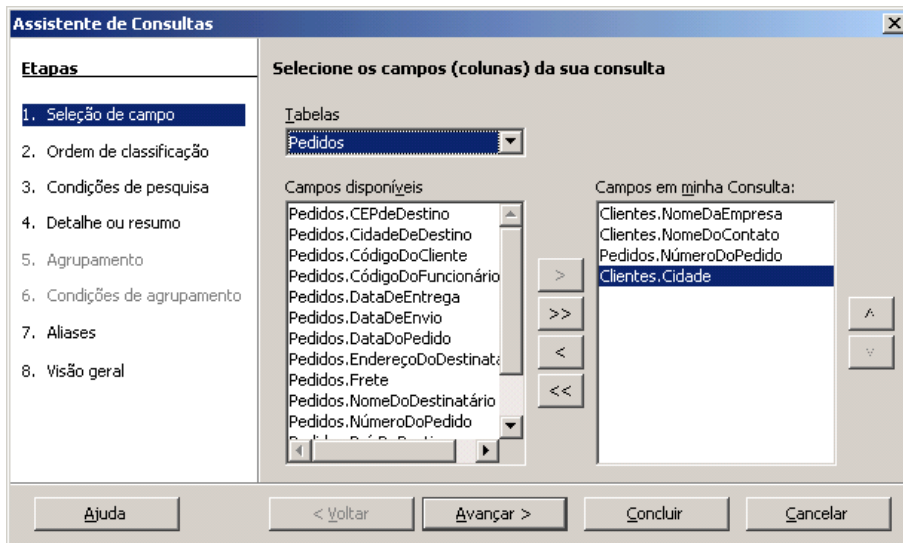
Para facilitar o entendimento, usaremos o exemplo das informações de Clientes e Pedidos para a criação da nossa consulta, com uma pequena modificação em relação ao critério de retorno de informações. Buscaremos, neste caso, apenas as informações dos clientes da cidade de Madrid.

Primeiro, clique em **Usar assistente para criar consulta...** .

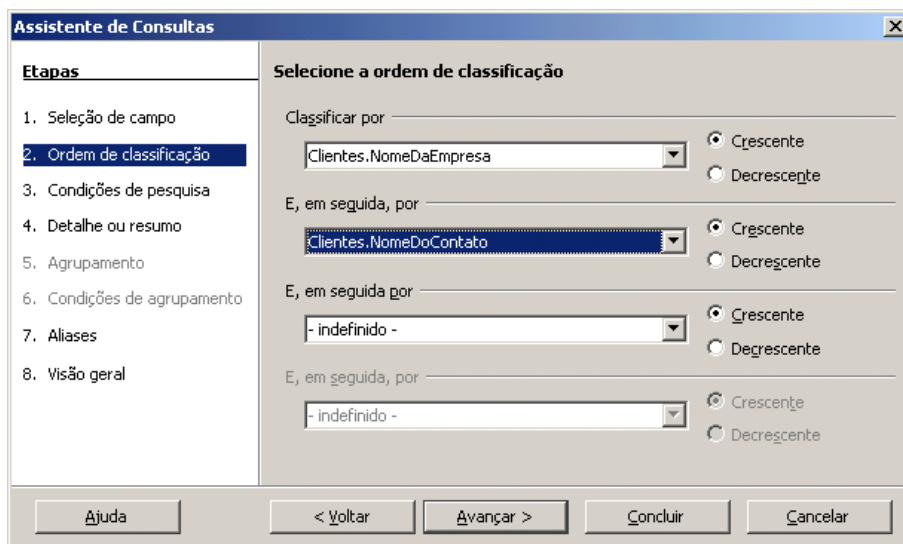


Será aberto o assistente, com a tela inicial de escolha dos campos que serão incluídos na consulta. Veja que, primeiro, você escolherá a tabela do campo e, depois, o campo propriamente dito.

Ao final das escolhas correspondentes ao exemplo, teremos:



Clique em **Avançar**.

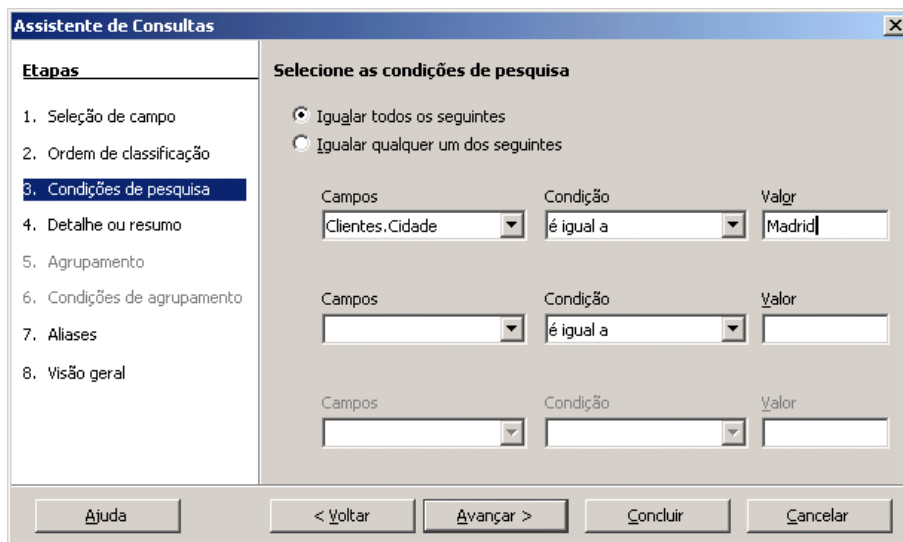


Na próxima tela, teremos a definição dos critérios de ordenação, onde definiremos uma ordenação básica.

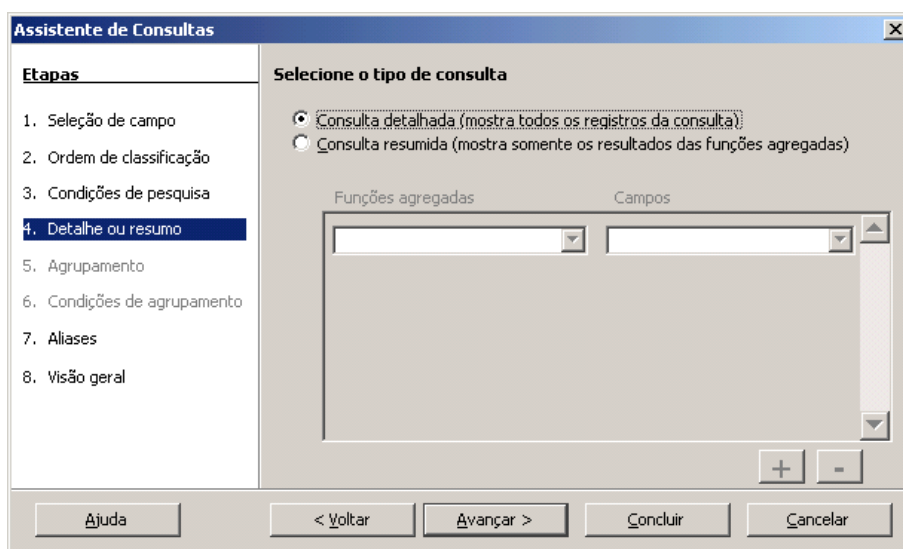
Clique em **Avançar**.

Na próxima tela, teremos a inclusão das condições de pesquisa, ou seja, dos critérios de refinamento da nossa consulta. É aqui que faremos com que a nossa consulta fique restrita apenas aos clientes da cidade de Madrid, indicando que o campo Clientes.Cidade deverá ser igual ao conteúdo “Madrid”.

A opção **Igualar todos os seguintes**, significa que os registros retornados terão que, obrigatoriamente, obedecer a todos os critérios indicados na lista. A opção **Igualar qualquer um dos seguintes**, indica, por sua vez, que os registros retornados poderão corresponder a apenas uma das características listadas nos critérios abaixo.

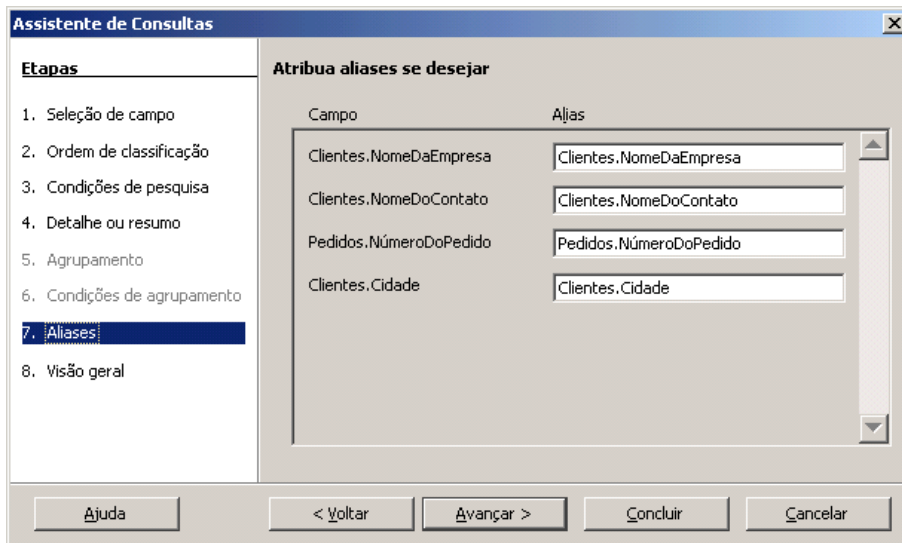


Clique em **Avançar**.



Na tela seguinte, poderemos incluir funções para contabilizar resultados de funções agregadas. As funções agregadas serão apresentadas em exemplos na definição de consultas através da construção de cláusulas SQL, mais adiante. Para o nosso exemplo, vamos manter a opção **Consulta detalhada (mostrar todos os registros da consulta)**.

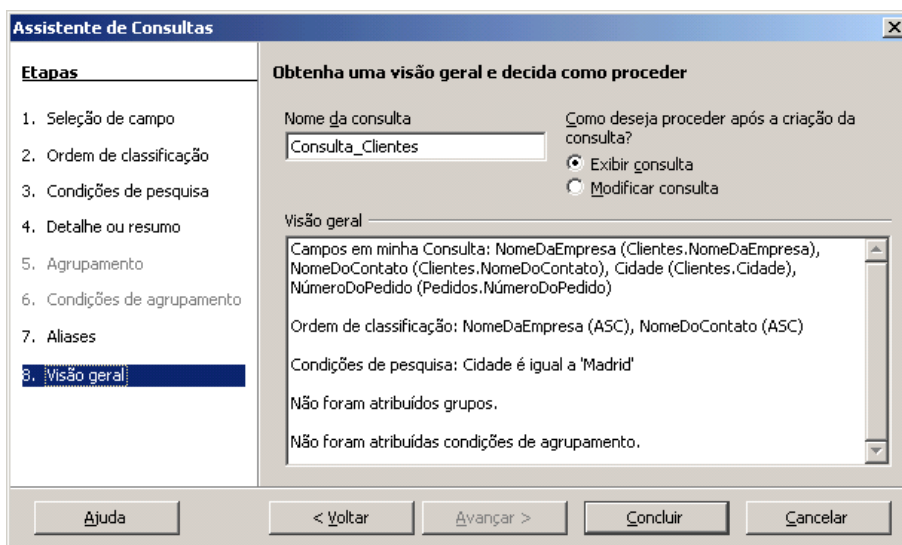
Clique em **Avançar**.



A tela seguinte apresenta a definição dos Aliases, que nada mais são do que nomes de campos mais familiares ao usuário. A redefinição do nome do campo é opcional, no entanto, recomendamos a manutenção da nomenclatura do sistema para a identificação total da origem dos dados.

Defina, então, se necessário, os novos nomes dos campos e clique em **Avançar**.

Por último, é apresentada uma visão geral da consulta, onde podemos rever todos os atributos definidos e alterá-los, caso seja preciso (em **Modificar consulta**).



Para finalizar a operação e obter os resultados, vamos manter marcada a opção **Exibir consulta** e clicar em **Concluir**.

O resultado será mostrado numa tela de exibição de consulta.

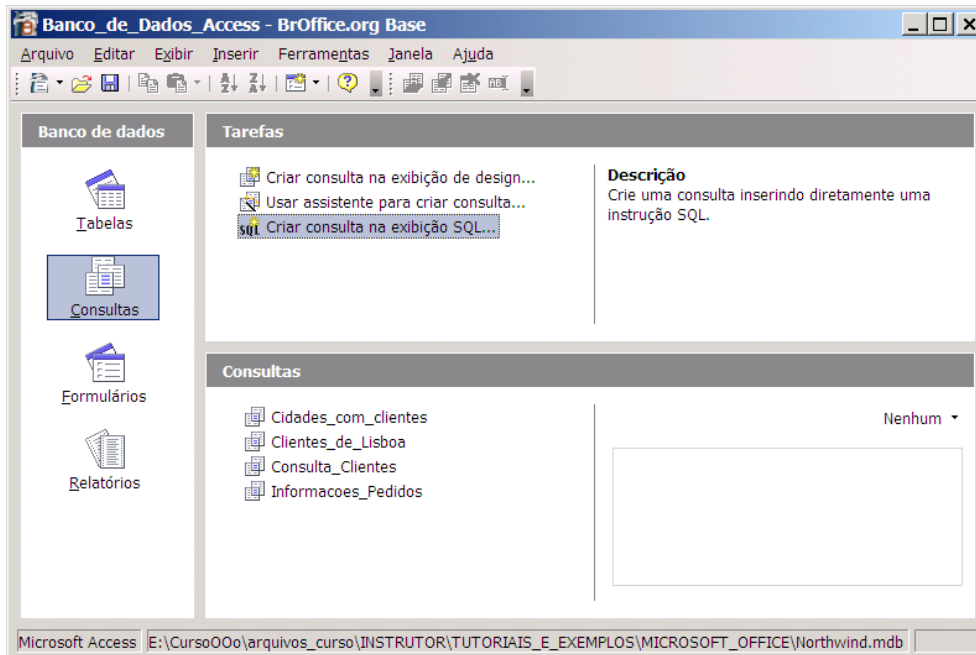
| NomeDaEmpresa | NomeDoContato | Cidade | NúmeroDoPedido |
|----------------------|---------------|--------|----------------|
| Bólido Comidas prepa | Martín Sommer | Madrid | 10825 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10818 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10832 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10494 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10831 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10495 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10830 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10496 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10829 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10497 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10828 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10498 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10827 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10499 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10833 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10822 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10787 |
| Bólido Comidas prepa | Martín Sommer | Madrid | 10819 |

Registro 1 de 35 *

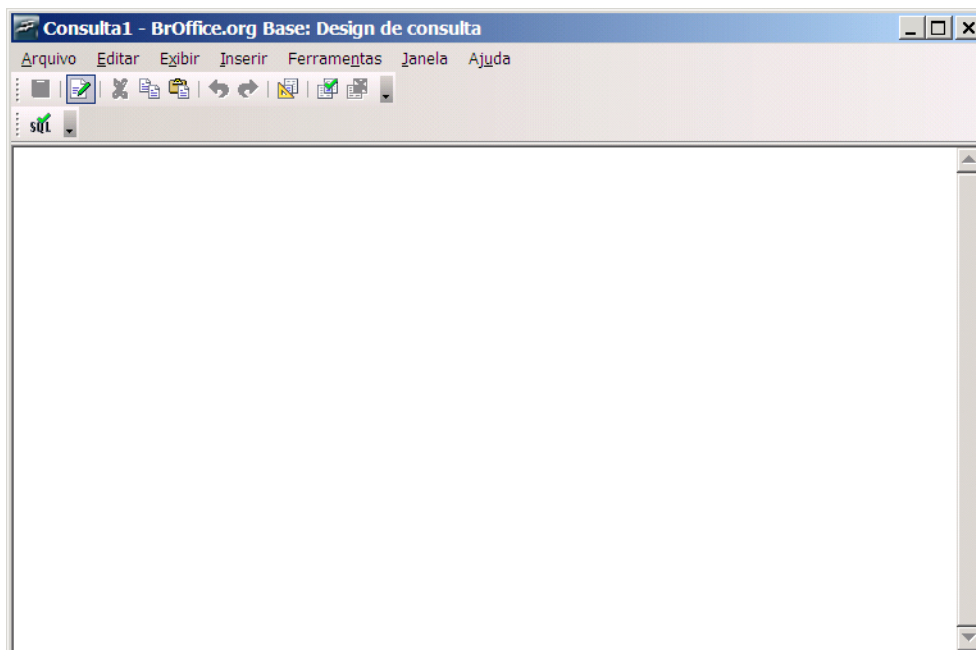
Criar consulta na exibição SQL...

Por último, temos a alternativa de criar a consulta através da inserção de comandos SQL. Essa alternativa exige conhecimento da sintaxe de criação de comandos SQL, no entanto, é a forma mais poderosa e flexível de criação de consultas.

Na seção de Consultas, clique em **Criar consulta na exibição SQL...**

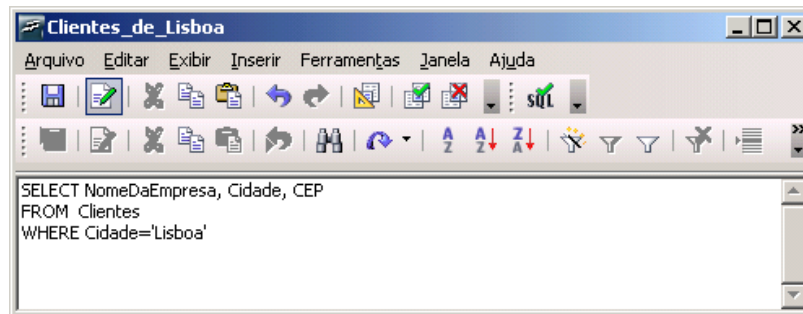



Será aberto o editor de consultas SQL, que nada mais é do que um espaço para a inserção dos comandos SQL.

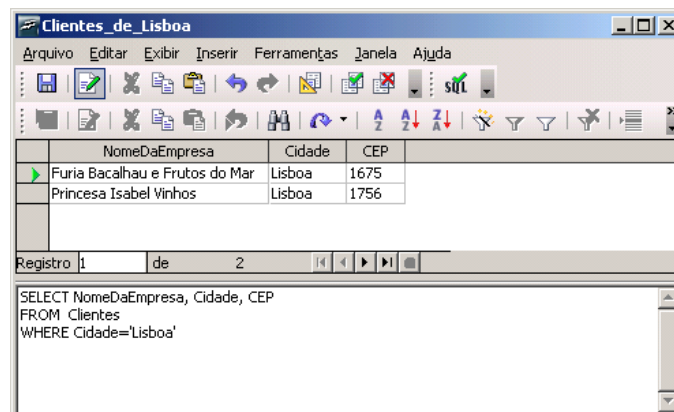


Se quiséssemos consultar as informações dos clientes de Lisboa, por exemplo, teríamos que escrever a seguinte consulta:

```
SELECT NomeDaEmpresa, Cidade, CEP  
FROM Clientes  
WHERE Cidade='Lisboa'
```



Finalizada a construção da consulta, o resultado pode ser exibido com o clique no botão , chamado Executar Consulta. O resultado é exibido acima da área de edição:



2.4 Diferenças entre Consultas e Exibições

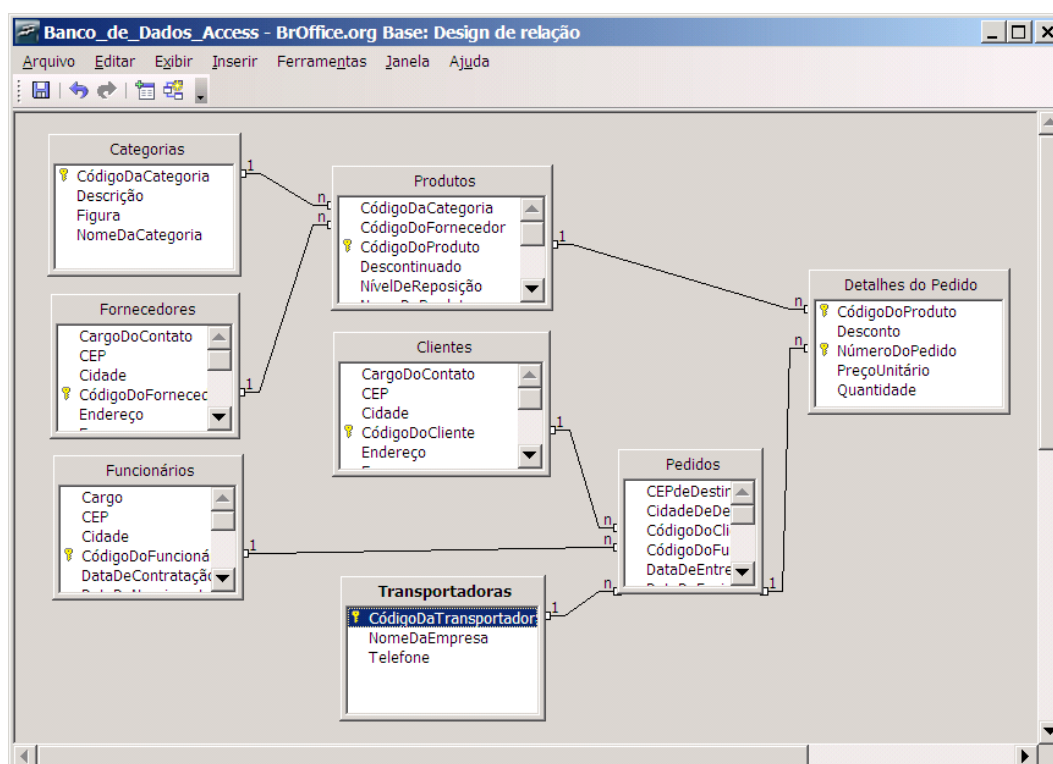
Entre as exibições e as consultas existem diferenças significativas:

| Propriedade | Exibição | Consulta |
|---|---|-----------------|
| Possibilidade de consultas parametrizadas | Não | Sim |
| Possibilidade de alterações posteriores no BrOffice.org | Não | Sim |
| Salvamento na seção Consultas do arquivo .mdb | Sim, se foi criada no Access. Não, se foi criada no BrOffice.org | Não |

2.5 Relacionamentos

Relacionamentos são ligações lógicas entre as tabelas, identificando comportamentos derivados dos processos de trabalho do usuário. Os relacionamentos auxiliam a organização das informações, minimizam as redundâncias e erros de digitação e facilitam a recuperação de dados através de consultas. Para que possamos ter relacionamentos, é necessário identificar adequadamente as chaves primárias.

No nosso exemplo, para vermos os relacionamentos, vamos até o menu **Ferramentas > Relações**. Será aberta a tela abaixo:



Na tela de **Design da Relação** podemos identificar claramente as tabelas. As linhas que ligam as várias tabelas identificam os relacionamentos. Note que, na ponta da linha, temos o número 1 ou a letra n. Esses caracteres identificam o comportamento do relacionamento na tabela indicada. Vamos tomar como exemplo a relação da tabela Clientes e da tabela Pedidos.






Veja que, próximo a tabela Clientes, temos o número 1 e, na tabela Pedidos, temos a letra n. Se pensarmos no processo de trabalho, constataremos que um único cliente poderá registrar diversos pedidos. Para suportar essa característica do processo, o sistema deverá permitir o armazenamento de vários registros de pedidos de um único cliente.

A representação dessa característica é feita através do relacionamento entre a tabela Clientes e a tabela Pedidos, que pode ser lido como um relacionamento “1 para n”, ou seja, 1 registro da tabela Clientes pode ser referenciado por n registros da tabela Pedido. Seguindo essa lógica, podemos estabelecer tipos de relacionamentos entre tabelas:

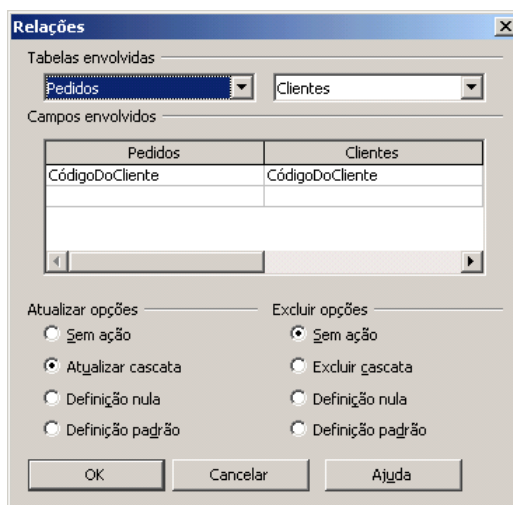
| | |
|----------|--|
| 1 para 1 | O relacionamento 1 para 1 é menos usual, pois normalmente, os dados podem ser armazenados em uma única tabela. Eventualmente, por critérios técnicos ou pela organização do sistema, utilizamos esse tipo. |
| 1 para n | É tipo de relacionamento mais comum. Permite fácil identificação, organização e recuperação dos dados. |
| n para n | Normalmente, identificado ainda no momento da análise do sistema, anterior à criação do banco de dados propriamente dito. Quando identificadas, essas relações normalmente são substituídas por relações 1 para n. |

Criando Relacionamentos...

A barra padrão da tela de Design da Relação possui cinco ícones:

| Ícone | Descrição |
|---|--|
|  | Salva o esquema de relações da tela. Além das relações propriamente ditas, é feito o salvamento da apresentação visual do esquema de relações. |
|  | Desfaz uma operação. |
|  | Refaz a operação desfeita. |
|  | Adiciona uma nova tabela ao esquema. |
|  | Permite a criação de novas relações. |

Ao clicar no ícone  (Nova Relação), é aberta a tela abaixo:



No exemplo acima, recriamos o exemplo da relação das tabelas Pedidos e Clientes. Note que, na tabela Pedidos, existe o campo CódigoDoCliente, que é o campo que faz a ligação entre as duas tabelas. Selecionando este campo na lista de Campos Envolvidos, estabelecemos a relação entre as duas estruturas.

Temos ainda as opções de atualização e as opções de exclusão. Exemplificando através da relação Pedidos/Clientes, teríamos o seguinte comportamento.

Atualização

| | |
|-------------------|--|
| Sem ação | Um registro atualizado na base de Clientes não afeta os registros da base Pedidos. |
| Atualizar cascata | Um registro atualizado na base de Clientes atualiza os registros correspondentes na base Pedidos, caso necessário. |
| Definição nula | Um registro atualizado na base de Clientes atualiza os registros correspondentes na base Pedidos como "IS NULL". |
| Definição padrão | Caso uma chave primária seja modificada na tabela Clientes, os valores correspondentes na tabela Pedidos são marcados com os valores padrões dos campos que compõe a chave primária. |

Exclusão

| | |
|-------------------|--|
| Sem ação | Um registro excluído na base de Clientes não afeta os registros da base Pedidos. |
| Atualizar cascata | Um registro excluído na base de Clientes exclui os registros correspondentes na base Pedidos. |
| Definição nula | Um registro excluído na base de Clientes atualizaria os registros correspondentes na base Pedidos como "IS NULL". |
| Definição padrão | Caso uma chave primária seja excluída na tabela Clientes, os valores correspondentes na tabela Pedidos são marcados com os valores padrões dos campos que compõe a chave primária. |

No caso do nosso exemplo, podemos escolher as opções **Atualizar cascata** para atualização e **Sem ação** para exclusão. Assim, definiremos que os registros das tabelas serão sempre atualizados quando houver uma alteração na chave primária e que não serão excluídos os registros das demais tabelas da base de dados caso um registro de uma tabela seja excluído.

3 Créditos e Agradecimentos

3.1 Créditos

Autor: Gustavo Buzzatti Pacheco

Coordenador de Documentação: Gustavo Buzzatti Pacheco <gbpacheco@broffice.org>

Coordenador Geral: Claudio Ferreira Filho <filhocf@broffice.org>

3.2 A Fazer

Detalhar, para arquivos .mdb, a manipulação de formulários e relatórios no BrOffice.org.

3.3 Agradecimentos

O BrOffice.org reconhece e agradece a todos os membros da comunidade, em especial aos mencionados em <http://www.openoffice.org/welcome/credits> e <http://www.broffice.org.br/creditos>.

No Brasil, a comunidade está representada em <http://www.broffice.org>.

3.4 Histórico do Documento

| Versão | Contribuição | Responsável | Contato | Data |
|--------|--------------|--------------------|--|----------|
| 0.9 | Autoria | Gustavo B. Pacheco | gbpacheco@BrOffice.org | 25/06/06 |