

Nesta seção você encontra artigos para iniciantes na linguagem Delphi

## Introdução ao Firebird e IBExpert

Veja como instalar o servidor e gerenciar bancos

Indispensável a todos os sistemas de pesquisas e cadastros, o banco de dados é o responsável por manter e organizar todas as informações recolhidas e geradas por um sistema. Em sua estrutura, o banco de dados mantém organizada uma série de conjuntos de informações, tais como: listas de produtos, detalhes de compras, vendas, informações de clientes entre inúmeras outras situações onde o banco de dados mantém os registros armazenados, possibilitando futuras consultas para o resgate das informações salvas.

Neste artigo da seção *Easy Delphi* veremos a instalação do SGBD (Sistema Gerenciador de Banco de Dados) Firebird 2.0 e também a instalação do IBExpert, uma das ferramentas mais poderosas e mais utilizadas para o gerenciamento de banco de dados Interbase/Firebird.

Finalizadas as instalações criaremos uma nova base de dados através do IBExpert onde iremos definir uma estrutura

de tabelas, campos e relacionamentos para armazenagem de dados. Ao final veremos a utilização dos principais comandos de linguagem *SQL* com significado mais adiante, para consultas, alterações, exclusão e edição dos registros armazenados na estrutura.

### Firebird: Conceito e instalação

O Firebird possui características robustas, é leve e poderoso ao mesmo tempo, sendo assim é capaz de atender ao mesmo tempo aplicações de um usuário único ou em uma rede de múltiplos acessos em nível corporativo.

Um único servidor Firebird poderá gerenciar várias bases de dados simultaneamente com múltiplas conexões, é um SGBD de necessidades mínimas de configuração e administração, é Open Source, ou seja, livre de licenças mesmo para uso comercial, permite a criação de procedimentos (Stored Procedures) que são processados internamente em sua



**Maikel Marcelo Scheid**

(maikelscheid@gmail.com)

é técnico em Informática com ênfase em Análise e Programação de Sistemas. Atua na área de Desenvolvimento de Softwares em Delphi para plataforma Win32 e .NET com banco de dados Firebird e MS SQL. É membro da Equipe Editorial ClubeDelphi.

camada servidora, criação de Triggers múltiplos por tabela, disparados para cada linha da tabela antes ou depois de Inserts, Updates ou Deletes com o intuito de gerar exceções, aplicar valores padrões e assegurar da integridade da base de dados.

Outra característica importante do Firebird são os *Generators/Sequences* (Geradores) que permitem colunas do tipo auto-incremento, que numeram em ordem seqüencial e crescente as linhas da tabela, além de *Backups Online*, *Integridade Referencial* em *Cascata* com controle total de suas transações. É também um SGBD multi-plataforma (*Windows, Linux, Mac OS X, Solaris e HP-UX*) com exigências mínimas de *hardware* para a instalação do servidor, que poderá ser nas versões *Classic Server* (uma instância por conexão cliente), *SuperServer* (todas as conexões são manipuladas por um único módulo) e *Embedded Version* (versão embutida onde o sistema inteiro é uma única DLL para uso por um único usuário e uma única base de dados).

A instalação do Firebird é bastante simples, você poderá fazer o download do arquivo executável da instalação pelo endereço [www.firebirdsql.org](http://www.firebirdsql.org). Veremos neste artigo a instalação da versão 2.0 do SGBD, mas que poderá ser instalado também na sua versão 1.5. A instalação propriamente dita é toda em português com várias dicas e instruções durante o processo de instalação que ocorre em 12 etapas enumeradas a seguir:

- Etapa 1: Seleção do idioma da instalação; escolha a opção Português e confirme;
- Etapa 2: Apresentação da ferramenta no assistente de instalação, utilize o botão Seguinte para continuar a instalação;
- Etapa 3: Termo de Licença; após a leitura do termo, escolha a opção Aceito o Contrato e clique em Seguinte;
- Etapa 4: Informações sobre o Firebird, dicas e configurações da instalação e remoção;
- Etapa 5: Destino da instalação, que será instalada por default no caminho "C:\Arquivos de Programas\Firebird\Firebird\_2\_0\", aconselhável de ser mantido;
- Etapa 6: Componentes a ser instalados; mantenha as opções padrões da

instalação (Figura 1);

- Etapa 7: Criação de atalhos no menu/desktop;
- Etapa 8: Tarefas adicionadas a serem executadas; mantenha as opções default (Figura 2);
- Etapa 9: Resumo das configurações para instalação;
- Etapa 10: Instalação do Firebird 2.0;
- Etapa 11: Resumo e informações; resultado da instalação;
- Etapa 12: Início do processo para funcionamento do Firebird. Após esta etapa o serviço será iniciado automaticamente

em cada restart do Sistema Operacional. Conclusão da instalação;

Neste momento o SGBD já estará instalado no computador e seus recursos poderão ser utilizados a qualquer momento para criação/hospedagem de arquivos de armazenagem de dados.

## IBExpert: Conceito e instalação

O IBExpert é uma das ferramentas mais poderosas e mais utilizadas para o gerenciamento de banco de dados Firebird. Permite realizar todas as tarefas necessárias no suporte e na manutenção dos arquivos

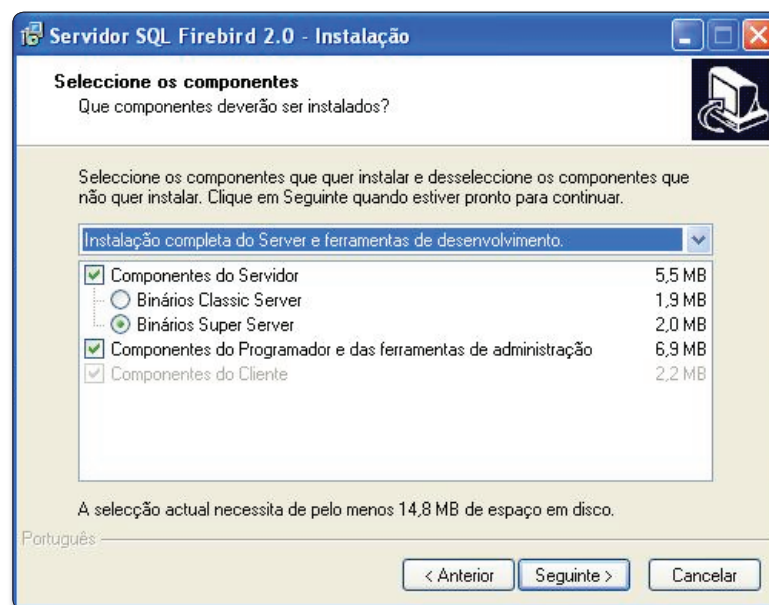


Figura 1. Componentes da instalação do Firebird

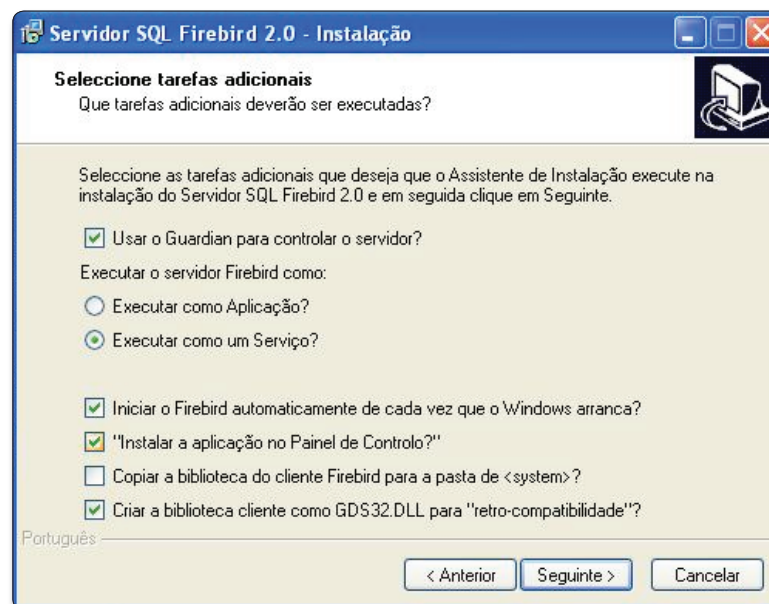


Figura 2. Tarefas adicionais a serem executadas junto ao Firebird

de banco de dados, possibilitando a criação e modificação de recursos, tabelas, campos, índices, entre outras funcionalidades, além da inserção e modificação das informações salvas na base.

Possibilita gerenciamento local ou remoto da base de dados, executando *scripts* e funções diversas. O *download* da ferramenta, que é fundamental para o DBA (responsável pela administração do banco de dados), poderá ser realizado pelo endereço [www.ibexpert.com](http://www.ibexpert.com), com a possibilidade de *download* de duas versões do *software*, uma que é *free* (porém com recursos limitados) e outra versão *shareware* (com todos os recursos num período de experiência de 45 dias).

A instalação do IBExpert (em Inglês) também ocorre de forma bastante sim-

ples, num processo dividido em 6 etapas e configuração. Veremos nesse artigo a instalação da versão 2007.09.25 do produto no processo numerado a seguir:

- Etapa 1: Apresentação da ferramenta no assistente de instalação, utilize o botão Next para continuar a instalação;
- Etapa 2: Escolha dos produtos a serem instalados. Nesta tela iremos deixar marcada apenas a opção IBExpert, ignorando a instalação das demais ferramentas (Figura 3);
- Etapa 3: Termo de Licença; após a leitura do termo, escolha a opção I accept the agreement e clique em Next;
- Etapa 4: Pasta de destino da instalação. Mantenha o caminho default que é apresentado pelo instalador (C:\Arquivos de Programas\HK-Software);

• Etapa 5: Breve resumo antes da instalação. Utilize o botão Install para continuar;

• Etapa 6: Instalação completa. Utilize o botão Finish para deixar o assistente;

Neste momento a instalação do IBExpert já está concluída e poderá ser utilizada para a criação e manutenção de arquivos de armazenagem de dados.

## Criando o primeiro banco de dados

Para iniciar a criação do primeiro banco de dados, abra o IBExpert e logo após clique sobre o menu *Database>Create Database*. Uma janela será exibida (Figura 4) para as configurações da nova base de dados. O item Server no IBExpert oferece as opções Remote, que irá criar a fonte de dados em uma máquina da rede que está acessível ou a opção Local, usada para criar a fonte de dados nesta mesma máquina.

Na opção *Database* defina o caminho onde a fonte de dados será armazenada, atribua a fonte o nome "banco.fdb". Em relação à autenticação do usuário e senha iremos utilizar as informações específicas do Firebird informando o usuário "SYSDBA" e a senha "masterkey" que serão os mesmos para todas as instalações padrão do SGBD.

Para a opção *Page Size* (tamanho da página) utilize o valor 4096, para *Charset* (codificação de caracteres) utilize WIN1252, que permitirá o uso de caracteres especiais ao inserir um registro e ainda na opção *SQL Dialect* (padrão de consultas e declarações SQL) defina como *Dialect 3*. Utilize o botão OK para continuar a criação da fonte de dados.

## Registrando o banco de dados

Após a criação da nossa fonte de dados precisamos registrá-la no IBExpert. Se você deixou habilitada na tela anterior a opção "Register database After Creating" de forma automática surgirá o diálogo de registro do BD. Caso a opção tenha permanecido desmarcada, utilize o menu *Database>Register Database* para exibir a caixa de configurações (Figura 5).

No registro de um banco de dados, semelhante à criação precisamos definir o *Server* onde está o arquivo do banco, definindo então como *Local*. Ao lado temos uma configuração para definir a versão do servidor que vamos utilizar,

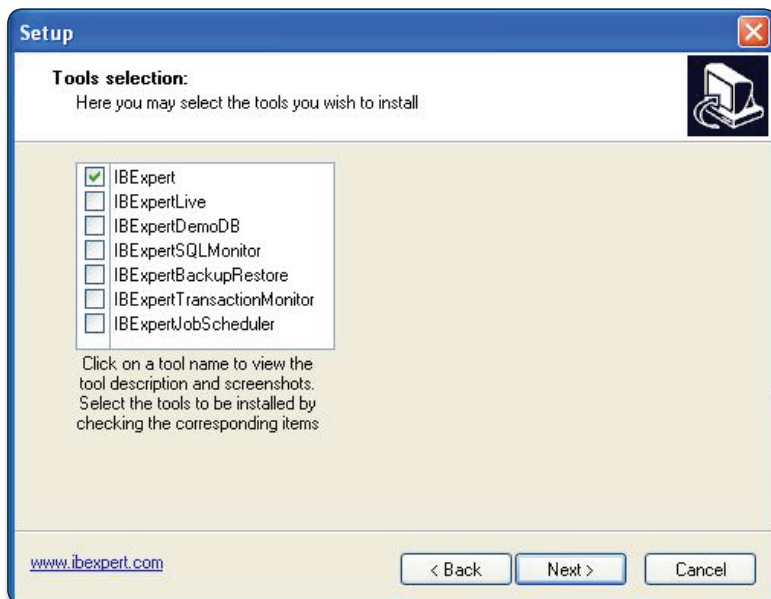


Figura 3. Configuração das ferramentas a serem instaladas

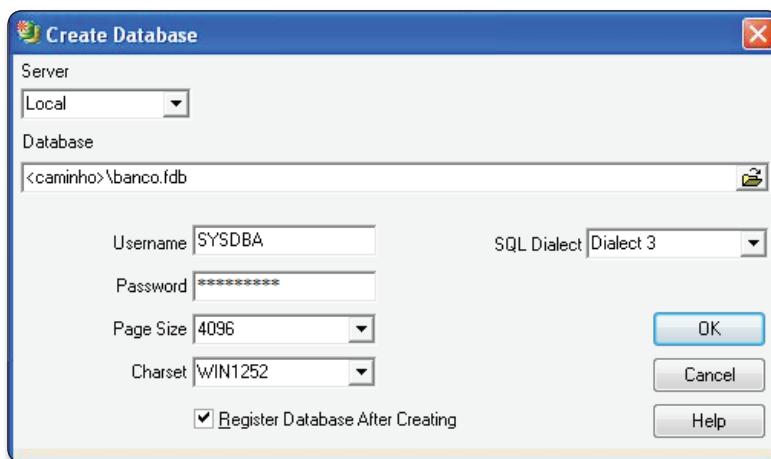


Figura 4. Criando fonte de dados



onde iremos definir *Firebird 2.0* que foi o SGBD instalado.

No caminho da base de dados defina o local onde o banco foi criado apontando para o arquivo "banco.fdb". Defina um nome (*Alias*) para o registro que será utilizado para identificar e diferenciar este dos demais registros de bancos que poderá fazer.

Da mesma forma que na criação da base de dados precisamos informar os usuário e senha para autenticação do registro ("SYSDBA" e "masterkey" respectivamente) e selecionar novamente a tabela de caracteres (WIN1252). Confirme o registro e perceba que o nome da base registrada aparecerá na paleta *Database Explorer* do IBExpert. Para conectar-se a base de dados registrada dê um clique duplo sobre o mesmo.

**Nota:** O IBExpert, em suas configurações, permite que seja utilizado no idioma Português. Para realizar a troca da linguagem vá até o menu *Options>Environment Options* e selecione a guia *Preferences*. Nela localize a opção *Interface Language*. Encontre o idioma *Portuguese* e confirme.

## Criando tabelas no banco de dados

Após conectar-se ao BD, observe que estão listados vários níveis de objetos, e que dentre eles temos um nível chamado *Tables* (Tabelas). Clique com botão direito do mouse sobre *Tables* e escolha a opção *New Table* que dará origem a interface de criação e configuração da primeira tabela do banco de dados. Criaremos duas tabelas no banco, *Funcionários* e *Cargos*.

Ao acionar o comando *New Table* note que a direita do aplicativo é mostrado um campo com a palavra *New\_Table*. *New\_Table* é a sugestão de nome de tabela que o IBExpert nos dá. Então clique neste campo e digite o nome da primeira tabela que criaremos: *Funcionarios*.

Logo abaixo podemos ver uma grade com apenas um registro. Cada registro nessa base é um campo que estamos criando na tabela. Como ainda não criamos nenhum campo a indicação do IBExpert é *New\_Field*. No lugar de *New\_Field* digite o nome do campo e as colunas subsequentes preencha-as de

acordo com a situação.

Ex. Para o primeiro campo digite *ID\_FUNCIONARIO* na coluna *Field Name*. Em *Field Type* escolha *Integer* em seguida localize a coluna *AutoInc* e marque-a.

Logo após marcar como *AutoInc* uma caixa de diálogo é aberta para configuração do auto-incremento (**Figura 6**). Na aba *Generator* marque a opção *Create Generator*. Em seguida dê um nome ao *Generator* e um valor inicial, normalmente usa-se 0.

Para que o *Generator* funcione automaticamente usamos outro recurso do banco Firebird: *Triggers*. Então selecione a aba *Trigger* e ative a opção *Create Trigger*. As *Triggers* são também chamados de gatilhos. Esse gatilhos são disparados mediante a um evento que acontece na tabela, nesse caso *Before Insert*, ou seja, toda vez que um registro for inserido nessa tabela a *Trigger* é chamada e o *Generator* ativado.

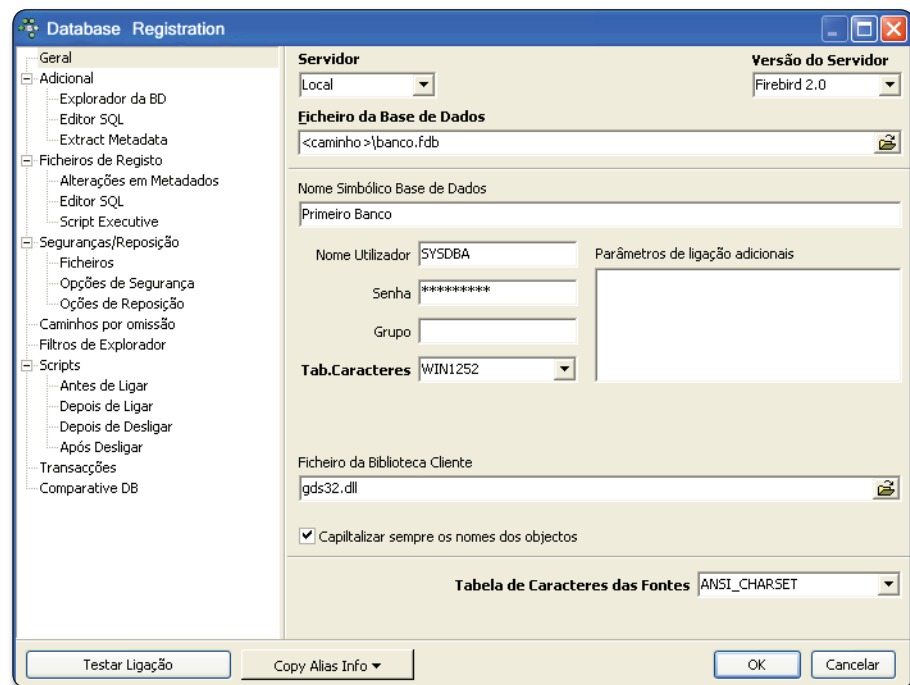
Marque a opção *Create Trigger* e perceba que o próprio IBExpert cria a instrução SQL que será usada para incrementar o campo *ID\_FUNCIONARIO* conforme a **Listagem 1**. Apenas confirme a caixa de diálogo.

Há também outra providência a se tomar. Note, nas especificações do campo *ID\_FUNCIONARIO*, que ele é um *Primary Key*, ou seja, chave primária. Nesse caso para tornar um campo par-

te da chave primária clique na coluna PK e pressione a barra de espaços. Um ícone em forma de chave será exibido na coluna.

**Nota:** Na estrutura de criação de tabelas de um banco de dados não é permitido o uso de caracteres especiais. Crie todos os campos sem acentuação. Ex: FUNCIONARIO, ENDE-RECO, etc.

**Nota:** A finalidade de uma chave primária é evitar entradas de dados em redundância garantindo desta forma a integridade dos dados, como por exemplo dois ou mais funcionários com o mesmo CPF cadastrados no sistema. Uma *Primary Key* pode também ser composta por mais de um campo. Ex: não podemos permitir que um mesmo produto entre duas ou mais vezes na mesma nota fiscal, nesse caso os campos da chave poderiam ser *COD\_NOTA* e *ITEM\_NOTA* ou algo similar. Para sanar o problema, caso o usuário final realmente comprei dois produtos iguais, criamos um campo que diferencie o registro no banco de dados, como por exemplo, *Quantidade*.



**Figura 5.** Registrando o banco de dados

Note que ao marcar a coluna como chave primária automaticamente a opção coluna *Not Null* é habilitada. Isso acontece porque um campo que faz parte de chave primária não pode nunca ser nulo, ou seja, precisa ser preenchido com algum valor. A estrutura completa de ambas tabelas no sistema pode ser vista em **Tabela 1** para *Funcionarios* e **Tabela 2** para *Cargos*. Crie os demais campos da sua tabela, observando apenas a alteração do tipo de dados (*Data Type*)

**Nota:** Na estrutura do campo *ID\_FUNCIONARIO* marque a opção *Not Null* para o campo *ID\_CARGO*, veremos a seguir como criar um relacionamento entre ambas tabelas.

Deixe sua tabela semelhante à estrutura exibida no **Figura 7**. Finalizada a criação dos campos da tabela localize no topo do aplicativo um ícone de raio com a função de compilar e criar a tabela (*Compile F9*). Na janela que irá surgir durante a com-

pilação selecione a opção *Commit* para concretizar a inclusão.

Após criada a tabela *FUNCIONARIOS* utilize o mesmo processo para criação da tabela *CARGOS*, compile e confirme a criação da mesma.

## Criando um relacionamento, Foreign Key

Uma *Foreign Key* (Chave Estrangeira) é um relacionamento entre duas tabelas de um banco de dados. O relacionamento é criado em uma prática chamada “pai para filho” onde uma chave primária da tabela pai é referenciada por um campo da tabela filho criando dessa forma a chave estrangeira. No nosso exemplo iremos relacionar o campo *ID\_CARGO* da tabela *CARGOS* (tabela pai) com o campo *ID\_CARGO* da tabela *FUNCIONARIOS* (tabela filho).

Para criar o relacionamento, abra a tabela *FUNCIONARIOS* com um clique duplo e observe que todos os campos estão sendo listados no *IBExpert*. Selecione a guia *Constraints* onde novas guias serão listadas, e entre estas localize e selecione a guia 2, *Foreign Keys*, exibindo um espaço em branco onde o relacionamento será criado.

Utilizando o botão direito do mouse sobre a área branca, selecione a opção *New foreign key*. Configurando o relacionamento mantenha o *Constraint Name* como sugerido pelo próprio *IBExpert*. Na próxima coluna, *OnField*, deverá ser informado em qual campo da tabela de *FUNCIONARIOS* vamos criar o relacionamento, selecionando o campo *ID\_CARGO*. Na coluna *FK Table* localize a tabela com a qual deseja relacionar o campo

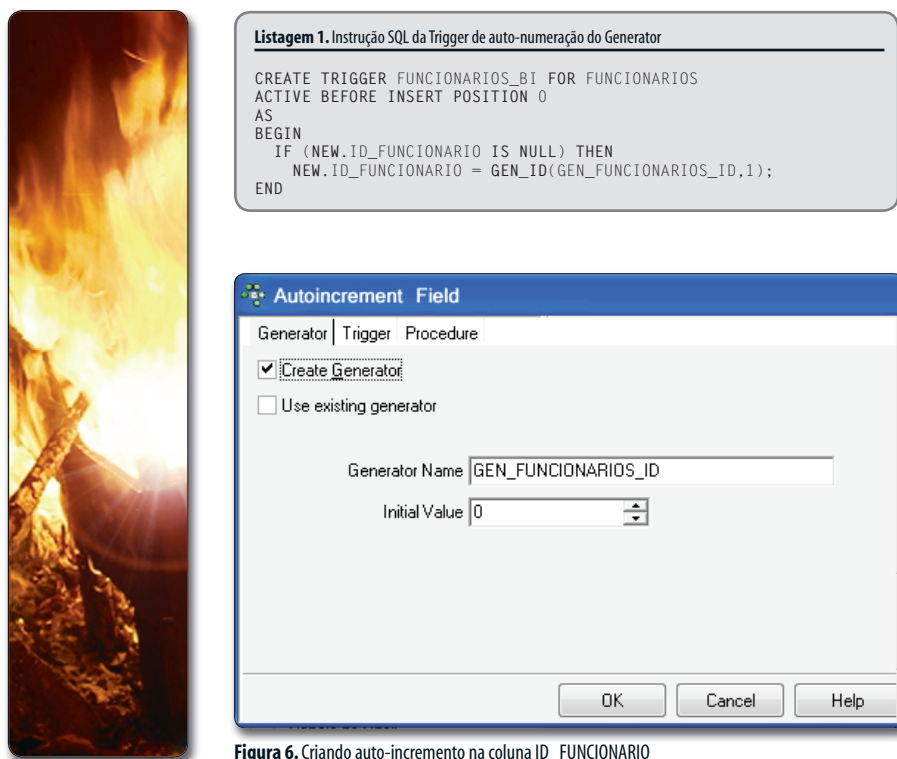


Figura 6. Criando auto-incremento na coluna *ID\_FUNCIONARIO*

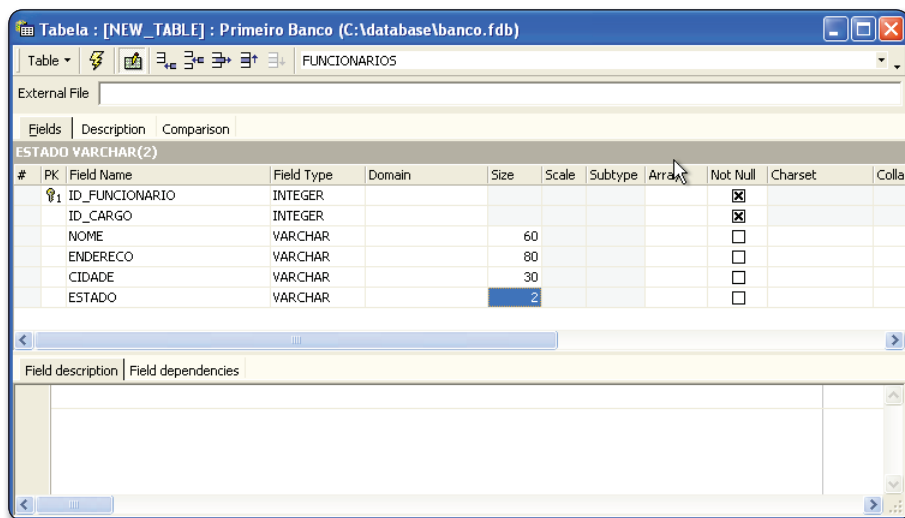


Figura 7. Tabela de *FUNCIONARIOS*

Campo	Tipo/Tamanho
ID_FUNCIONARIO	Integer(AutoInc) – Primary Key
ID_CARGO	Integer – Not Null
NOME	Varchar(60)
ENDERECO	Varchar(80)
CIDADE	Varchar(30)
ESTADO	Varchar(2)

Tabela 1. Estrutura da tabela de *Funcionários*

Campo	Tipo/Tamanho
ID_CARGO	Integer(AutoInc) – Primary Key
CARGO	Varchar(30)

Tabela 2. Estrutura da tabela de *Cargos*

selecionando, neste caso *CARGOS*, onde automaticamente na coluna *FK Field* já será listado o campo *ID\_CARGO* (**Figura 8**). Utilize novamente o ícone *Compile* ou *F9* para criar o relacionamento. Para confirmar selecione o botão *Close* na janela que irá aparecer.

Ao voltar para a guia *Fields* note agora que o campo *ID\_CARGO* foi marcado também com o ícone de uma chave, porém na coluna *FK*.

### Inserindo/visualizando registros direto na tabela

Para inserir novos registros ou visualizá-los basta que a tabela seja exibida e sua aba *Data* esteja selecionada. Automaticamente a lista de informações já será carregada. Clicando sobre o título de uma coluna é possível fazer ordenações e pesquisa das informações cadastradas.

Abra a tabela *CARGOS*, selecione a aba *Data* e digite no espaço reservado da coluna o cargo "Programador". Ignore o campo *ID\_CARGO* deixando-o sem informação, a qual será preenchida automaticamente pelo auto-incremento criado anteriormente para esta coluna. Utilize o símbolo "V" no topo da interface

(*Commit Transaction*) para salvar o registro. Observe que o campo *ID\_CARGO* assumiu o valor 1. Proceda desta maneira o cadastro de mais alguns cargos.

Selecione a tabela *FUNCIONARIOS* para inserção de registros, posicione-se na aba *Data* e clique sobre o campo *ID\_CARGO*. É difícil decorarmos os códigos dos registros, mas precisamos que o código de um cargo já cadastrado seja informado nessa coluna. Para não precisarmos ir até a tabela *CARGOS* pesquisar o código do cargo que desejamos inserir basta que sejam dados dois cliques sobre a coluna e verificar que uma lista de cargos é aberta, permitindo a pesquisa por qualquer um dos campos que forem selecionados. Para salvar o registro a tabela basta selecionar o campo desejado e pressionar a tecla *Enter*. Faça o cadastro de alguns funcionários para vermos a seguir a utilização dos principais comandos de *SQL*.

### Executando comandos SQL

*SQL* (Linguagem de Consulta Estruturada) é uma linguagem fácil e simples de pesquisa própria de banco de dados, mantendo um grande padrão entre to-

dos os demais tipos de bancos de dados (*MSSQL*, *MySQL*, *Oracle*, *FB* etc). Veremos nessa estrutura 4 formas diferentes da utilização desta linguagem, simulando as principais ações que são realizadas em banco de dados.

Abra no *IBExpert* o editor de comandos *SQL* através do menu *Tools>SQL Editor* onde deverão ser digitadas as instruções de *SQL* que veremos a seguir. Para executar a instrução e observar os resultados (em caso de pesquisa) utiliza-se o ícone *Play* (*Execute – F9*). Em casos de inclusão, exclusão ou alteração de registros, além de utilizar o botão *Execute* deverá utilizar também o ícone "V" (*Commit Transaction*) para que a ação seja persistida ao banco.

#### 1 - Consulta das informações

A pesquisa das informações cadastradas em um banco de dados é uma das transações mais utilizadas. É através deste método que conseguimos resgatar tudo o que está salvo na estrutura do mesmo. Podemos criar várias situações e formas diferentes de realizar uma pesquisa das informações, veja a seguir algumas utilizações:





*Pesquisa simples:* digite no *SQL Editor* a seguinte instrução de código *SQL* e execute a mesma. Estamos selecionando e exibindo todos os registros da tabela de funcionários sem qualquer restrição:

```
select * from funcionarios
```

*Pesquisa parametrizada:* na seguinte instrução de código *SQL* iremos definir uma condição (parâmetro) para a pesquisa, e somente os registros que atenderem a esta condição serão exibidos:

```
select * from funcionarios where
funcionarios.id_cargo = 1
```

Definimos na condição que somente os funcionários que forem pertencentes ao cargo de código número 1 serão listados, ignorando todas as demais ocorrências.

*Pesquisa com relacionamentos:* as pesquisas com relacionamentos são realizadas com intuito de mostrar informações de duas ou mais tabelas, utilizando as chaves estrangeiras como condições. No código da **Listagem 2** selecionamos todos os funcionários cadastrados e seus respectivos cargos. Veja que na cláusula *Inner Join* estamos definindo quem será a tabela filho

(*FUNCIONARIOS*) e qual o campo que será relacionado a mesma de origem da tabela pai (*CARGOS*).

## 2 – Editando registros

As edições em geral são sempre de um registro específico, ou seja, um registro que atenda antes uma condição. Vejamos um exemplo prático onde iremos editar o nome de um funcionário, utilizando uma condição de que o código do mesmo deverá ser igual a 1. Nesta edição, um único registro (*ID\_FUNCIONARIO = 1*) será afetado e alterado:

```
update funcionarios set funcionarios.nome =
'MAIKEL SCHEID'
where
funcionarios.id_funcionario = 1
```

## 3 – Inserindo registros

A inserção de registros com certeza também está entre as transações mais utilizadas em bancos de dados. Sua funcionalidade é de armazenar novas informações nas tabelas. No exemplo a seguir faremos o cadastro de um novo funcionário utilizando comandos *SQL*:

```
insert into funcionarios (id_cargo, nome)
values (5,'PAULO')
```

Observe que ao passar os valores para serem inseridos na tabela de funcionários, precisamos tomar alguns cuidados em relação ao tipo de dados de cada campo. O *ID\_CARGO* é de valor inteiro, podendo ser informado somente o número correspondente ao cargo, já o campo *NOME* é do tipo *Varchar* exigindo que seu valor seja passado entre aspas simples.

## 4 – Excluindo registros

A exclusão de registros geralmente não é muito utilizada em bancos de dados. A metodologia adotada pela grande maioria dos programadores apóia-se em não excluir as informações cadastradas, mas sim deixando-as no banco de dados como forma de histórico e atribuindo a tabela uma coluna de *STATUS*, que para cada registro irá definir se o mesmo poderá ou não ser visível ao sistema. No exemplo veremos a exclusão de um registro de funcionários, o qual deverá atender a condição de *ID\_FUNCIONARIO = 6*:

```
delete from funcionarios
where funcionarios.id_funcionario = 6
```

Tome cuidado em código de exclusão para sempre utilizar um parâmetro, para não correr o risco de perder todas suas informações cadastradas.

A linguagem *SQL* possui uma grande variedade de comandos e requer um estudo aprofundado. O que vimos nessa seção do artigo foi apenas uma pequena introdução a linguagem. Recomendo de livros e artigos especializados em linguagem *SQL* para aquisição de mais conhecimento.

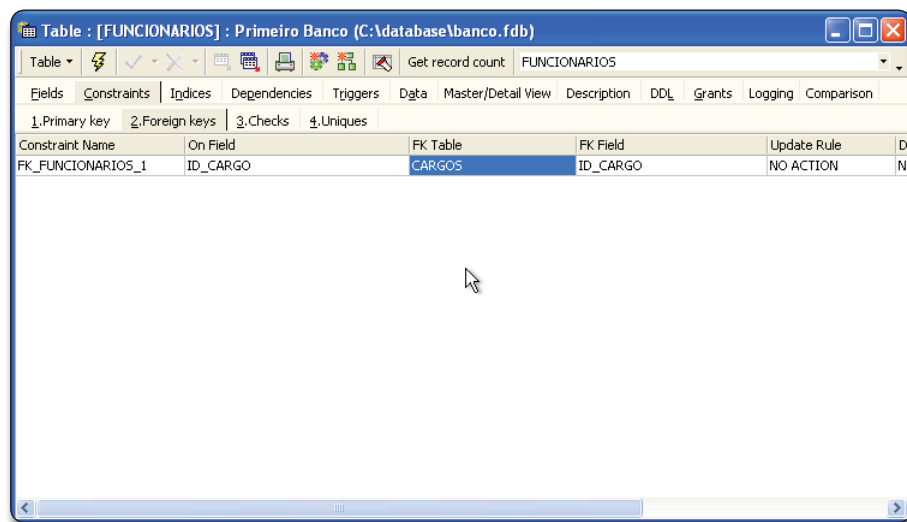
## Conclusão

Vimos neste artigo a instalação, configuração e uso do SGBD *Firebird* e do gerenciador *IBExpert*, o qual foi utilizado para criação de um banco de dados e estudo de algumas rotinas de linguagem *SQL*.

Embora breve, a introdução à linguagem *SQL* poderá lhe dar um rumo para aperfeiçoar ainda mais seus conhecimentos, pesquisar novos métodos, uso de parâmetros, *procedures*, implementação de domínios nos campos de tabelas entre outras várias situações que você irá afrontar ao usar um banco de dados para criação de um sistema. Um grande abraço e até a próxima. ●

**Listagem 2.** Exemplo de seleção de dados com mais de uma tabela

```
select
funcionarios.id_funcionario, funcionarios.nome,
funcionarios.endereco, funcionarios.cidade,
funcionarios.estado, cargos.cargo
from
cargos
inner join funcionarios
on (cargos.id_cargo = funcionarios.id_cargo)
```



**Figura 8.** Criando relacionamento de chaves entre tabelas