
Boas Vindas ao MS Access™

1º Tutorial - Access Por Resultados

Gil Jr

Historial de Revisões

Revisão 2

Data:15/12/2009

Revisão 1

Data:28/11/2009

Resumo

Access Por Resultados foca no objetivo do usuário: aonde se quer chegar ? Faz-se um estudo direcionado para o objetivo; agrega-se conhecimento conforme necessário; começa-se do simples e vai-se incrementando o banco de dados gradualmente, sem o objetivo de abraçar tudo de uma vez ! Aprender a essência da ferramenta fazendo banco de dados que dê o resultado pretendido; faz-se o protótipo e implementá-o, à princípio, despretenciosa e ingenuamente; depois, vai-se melhorando-o, conforme se apreende as técnicas.

Bem vindo à série de tutoriais *Access Por Resultados* !

Índice

1. Introdução	1
2. Sistema de Gerenciamento de Banco de Dados - SGBD	2
3. Estruturando um Banco de Dados	3
3.1. Etapa 1 - Determine o propósito de seu banco de dados	3
3.2. Etapa 2 - Determine as tabelas que você precisa	4
3.3. Etapa 3 - Determine os campos necessários	4
3.4. Etapa 4 - Determine os relacionamentos	5
3.5. Etapa 5 - Refine a estrutura do banco de dados	5
4. Conclusão	6
5. Recursos	6
6. Sobre o autor	6
Glossário	7

1. Introdução

Usar o MS Access é desafiante, mas prazeroso!

A interface do programa não se parece com outras que estamos mais habituados, como o MS Excel™ e o MS Word™.

Porém é o mais próximo que o usuário possa chegar do programador; em outras palavras, pode-se chegar mais longe ainda do que estamos acostumados com os editores e planilhas comuns, automatizando tabelas, consultas e relatórios, e melhor, reaproveitando-os.

Nos aventuramos nessa série de tutoriais tentando incutir-lhe o prazer de usar essa ferramenta: MS Access.

O MS Access é um banco de dados fabuloso, com um repertório de recursos vastíssimos - vários por nós ainda sendo descobertos...

Seus benefícios para o nosso dia a dia são inúmeros, e no caso específico da Operação Ecf, única alternativa que encontramos, frente algumas circunstâncias...

O limite de linhas de uma planilha no MS Excel é facilmente alcançado devido ao tamanho dos arquivos gerados pelo programa eECFc (usado pela fiscalização nas leituras dos equipamentos Emissores de Cupons Fiscais).

A administração de inúmeros e enormes arquivos é a oportunidade de usar recursos de um banco de dados.

Como dito, pode-se reutilizar planilhas, consultas, tabelas e relatórios no MS Access; pode-se importar artefatos de outros bancos de dados, ou seja, reaproveitar esforços, evitando o retrabalho, ou “reinventar a roda!” Nada de *selecionar / copiar / colar / modificar* que nos levam facilmente a erros. Nada de redigitar dados em editores de texto: nossos dados são guardados em banco de dados uma única vez e são reutilizáveis e inseridos nos lugares certos em relatórios, etiquetas... você escolhe!

Apreender a melhor técnica exige determinação, interesse e disciplina. Estamos prontos para aprimorarmos nossa solução mediante sugestões. Você tem uma página web (<http://code.google.com/p/accessbyresults/>) ao seu inteiro dispor; Mãos à obra; Interaja já !

Esse trabalho visa incentivar o uso MS Access.

Entramos diretamente no ponto: os exemplos serão extraídos do nosso dia a dia, especificamente da operação ECF.

Pouca teoria; definimos o que queremos e implementamos a solução.

Apreender as técnicas direcionando-nos para resultados.

Access por resultados !

2. Sistema de Gerenciamento de Banco de Dados - SGBD

Um banco de dados é uma coleção de dados. Há várias estratégias para organização das informações de modo a facilitar o seu acesso e manipulação.

Sistemas de Gerenciamento de Banco de Dados (Database Management System - DBMS) fornecem mecanismos para armazenamento, organização, resgate e modificação de dados; o acesso aos dados pelos usuários pode ser simultâneo e múltiplo.

Banco de dados relacionais são os mais usados atualmente. Vamos entendê-los e descobrir o porquê são a última palavra em termos de tecnologia em armazenamento de dados.

Várias empresas lançaram suas implementações de SGBD; produtos tais como Microsoft SQL Server, Oracle, MySQL, Infomix, PostgreSQL, Java DB (versão da Sun do Apache Derby), Firebird são largamente utilizados pelos programadores.

Cada um desses produtos oferecem driver, interface necessárias para nos comunicarmos com o banco de dados específico.

Esses *driver* normalmente são fornecidos para uma linguagem específica: Java, Python, C++, Pascal, Delphi, etc.

De posse desse driver pode o programador acessar os dados armazenados no banco de dados independente de sua representação interna.

Independente de sua representação interna ? o que isso significa nesse contexto.

Imagine que estamos armazenando dados em uma planilha do MS Excel, por exemplo; temos duas alternativas: ou acessamos os dados via o próprio programa MS Excel, ou teremos que descobrir a representação interna dos dados em uma planilha eletrônica para manipulá-los, algo difícil e penoso para programar: uma odisséia! se precisamos usar os dados num editor de texto, precisamos saber sobre compatibilidades de arquivos...um emaranhado se apresenta, e o problema é inicial e sem fim...

Acontece ainda que o MS Excel é uma planilha eletrônica e não um banco de dados...

A melhor solução, se quisermos organizar informações sistematicamente para reutilizá-las em diversas tarefas, é usar um SGBD, que é a própria essência do instrumento: gerenciar bancos de dados.

O MS Access é uma implementação de um SGBD da Microsoft™.

É uma ferramenta pronta para o uso; possui grande abstração: configurações internas, driver e outras peculiaridades, são automáticas; a manipulação dos dados é facilitada com o emprego de uma interface clara e intuitiva.

E como o MS access faz isso tudo ?

Existe uma linguagem neutra, comum a todos os SGBD, que opera todas as consultas que necessitamos, extraíndo-as de um banco de dados: SQL.

SQL (*Standard Queries Language*) é uma linguagem universal de procedimentos de consultas em bancos de dados relacionais.

Possui sintaxe simples, porém poderosa; algo como `SELECT * FROM Contribuinte`, que se constitui em um comando SQL que seleciona e retorna todos campos da tabela Contribuinte. Existem vários outros comandos, tais com INSERT, UPDATE, DELETE, que interagem com o banco de dados, independente de sua representação interna ou implementação. Retornaremos ao assunto mais adiante, em outro tutorial.

Através da SQL é que interagimos com o banco de dados programaticamente, resgatando informações necessárias aos relatórios, planilhas, e similares, que precisamos, assumindo certos critérios, para satisfazer os nossos requerimentos de consultas, inserções, etc.

O MS Access esconde do usuário comum as peculiaridades da linguagem SQL, buscando uma abstração que facilita a interação desse usuário com o banco de dados; porém, em se entendendo a linguagem, auferimos maior poder de consulta às nossas rotinas. Em última análise, passaremos a dar maior alcance aos nossos aplicativos. Portanto, investir no conhecimento da linguagem SQL é recomendável !

3. Estruturando um Banco de Dados

Aqui estão as etapas utilizadas no processo de estruturação de um banco de dados.

Cada etapa será discutida em maiores detalhes nos trabalhos remanescentes dessa série.


1. Etapa 1 - determine o propósito do seu banco de dados - isso ajuda a decidir quais dados serão armazenados pelo Ms Access.
2. Etapa 2 - determine as tabelas que você precisa - após decidir o propósito, divida as informações em assuntos diferentes (contextos diversos); cada assunto irá compor uma tabela do seu banco de dados.
3. Etapa 3 - determine os campos necessários - cada categoria de informação é chamada de campo e é mostrada como uma coluna independente na tabela.
4. Etapa 4 - determine os relacionamentos - decida como os dados de uma tabela se relacionam com os dados de outras tabelas.
5. Etapa 5 - refine a estrutura do banco de dados - verifique inicialmente se consegue obter os resultados desejados em suas tabelas. Caso necessário, reajuste a estrutura de seu banco de dados.

3.1. Etapa 1 - Determine o propósito de seu banco de dados

O primeiro passo na criação de um banco de dados é a determinação do objetivo do banco de dados e de como será utilizado. Isso indicará o tipo de informação a ser obtida. A partir daí, pode-se determinar os assuntos (tabelas) a serem introduzidos no banco de dados e quais os dados (campos) a serem armazenados, para cada assunto.

Nosso objetivo: Gerar o Termo de Leitura e cópia de Dados Gravados em Ecf, conforme figura abaixo:

Figura 1. Termo de Leitura e Cópia de Dados Gravados em Ecf

 SECRETARIA DE FINANÇAS DE RONDÔNIA		TERMO DE LEITURA E CÓPIA DE DADOS GRAVADOS EM ECF		NUMERO DO TERMO	
DADOS DO ESTABELECIMENTO					
RAZÃO SOCIAL:		I.E.:		CNPJ:	
ENDEREÇO:					
DADOS DO ECF					
MARCA:		MODELO:		VERSÃO S.B.:	
NÚMERO DE FABRICAÇÃO:					
AUTENTICAÇÃO DOS ARQUIVOS					
NOME DO ARQUIVO:					
MD5:		SHA1:			
NOME DO ARQUIVO:					
MD5:		SHA1:			
NOME DO ARQUIVO:					
MD5:		SHA1:			
NOME DO ARQUIVO:					
MD5:		SHA1:			
OBSERVAÇÕES ADICIONAIS					

3.2. Etapa 2 - Determine as tabelas que você precisa

Consulte o formulário anterior.

O próprio formulário nos dá dicas de tabelas candidatas; perceba as divisões internas no formulário.

Três seções distintas se apresentam: *Dados do Estabelecimento*, *Dados da Ecf* e *Autenticação de Arquivos*.

Três tabelas podemos então formar: *Contribuinte*, *Ecf* e *ArquivoAutenticado*.

São assuntos (contextos) diferentes, merecedores de tabelas separadas.

3.3. Etapa 3 - Determine os campos necessários

Transcrevendo os dados do formulário, teríamos os campos: *Razão Social*, *Inscrição Estadual*, *Cnpj* e *Endereço* para a tabela *Contribuinte*.

Os Campos *Marca*, *Modelo*, *Versão do Software Básico* e *Número de Fabricação* para a tabela *Ecf*.

E finalmente, os campos *Nome do Arquivo autenticado*, sua chave *MD5* e *SHA1*, repetidos quatro vezes, para a tabela *ArquivoAutenticado* (no máximo, por termo, só podemos autenticar quatro arquivos).

3.4. Etapa 4 - Determine os relacionamentos

O SGBD usa o relacionamento entre as tabelas para encontrar a informação solicitada.

Você define o relacionamento entre as tabelas, levando em consideração os conceitos de Direcionalidade, Multiplicidade e Tipo de Coleções.

No nosso exemplo analisemos a relação entre as tabelas *Contribuinte X Ecf*.

Cada Contribuinte possui nenhuma, uma ou várias Ecf (Relação zero ou Relação um-para-várias); Cada Ecf pertence, em determinado tempo, a apenas um Contribuinte (Relação um-para-um implementada pela inserção do campo *Codigocontribuinte* na tabela *Ecf*);

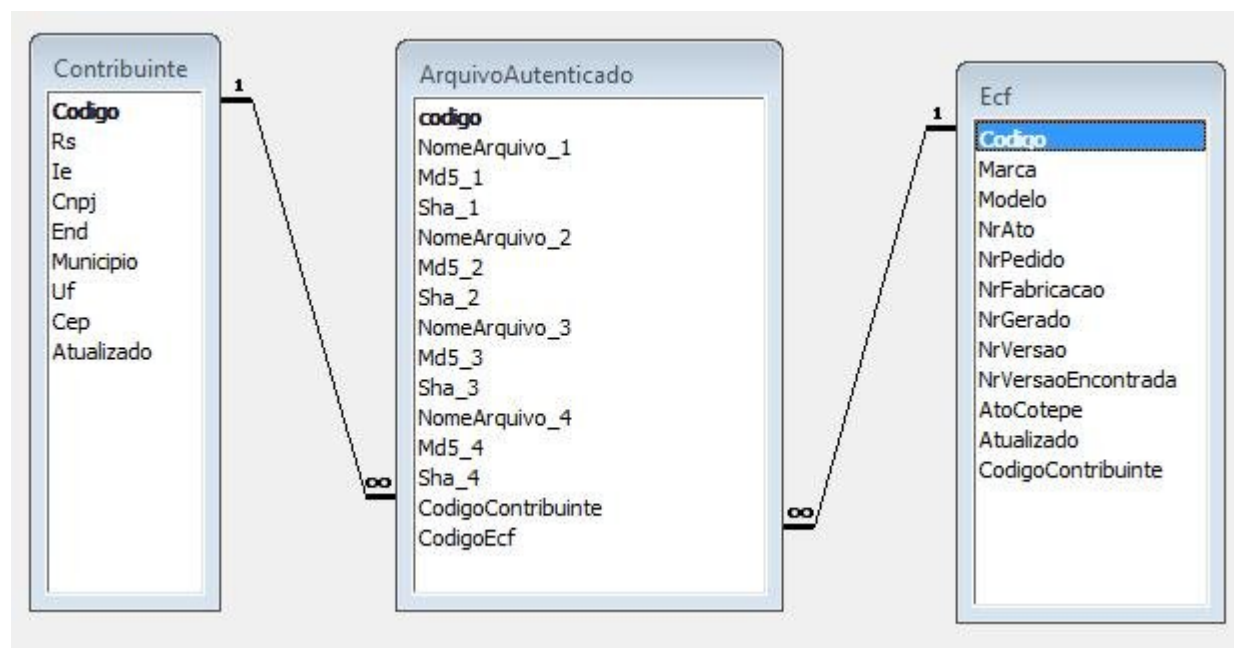
Quanto à relação entre as tabelas *Contribuinte X ArquivoAutenticado X Ecf*: cada arquivo autenticado refere-se apenas a um Contribuinte e a uma Ecf (implementação feita pela inserção dos campos *CodigoContribuinte* e *CodigoEcf* na tabela *ArquivoAutenticado*); porém, cada Contribuinte e cada Ecf podem ter mais de um arquivo autenticado - as relações entre as tabelas **Contribuinte X ArquivoAutenticado** e **Ecf X ArquivoAutenticado** são do tipo um-para-várias: perceba os símbolos de infinito e o número 1, identificando as relações de multiplicidade; 1(um) Contribuinte pode ter várias/infinitos ArquivoAutenticado(s); cada ArquivoAutenticado concerne a apenas 1 (uma) Ecf...

Pronto, decidido as relações entre as tabelas, vamos concretizá-las:

Nota

No 2º Tutorial da série entraremos em maiores detalhes.

Figura 2. Relacionamentos entre as tabelas



3.5. Etapa 5 - Refine a estrutura do banco de dados

Nessa fase acrescentamos alguns campos que achamos pertinentes.

Para a tabela *Contribuinte* acrescentamos os campos: *Município*, *Unidade da Federação* e *Cep*, por razões óbvias.

Para a tabela *Ecf* acrescentamos outros campos constantes de informações do arquivo extraído do Sitafe, quais sejam: *Número do Ato Cotepe*, *Número do Pedido de Uso/Cessação*, *Número Gerado* pelo Sitafe, *Número da Versão Atual* e *Número da Versão Encontrada* pela fiscalização, campos necessários para os trabalhos e relatórios futuros.

O campo *Atualizado* registra automaticamente a data de criação do registro, para rastreabilidade; constitui-se em boa prática na estruturação do banco de dados, assunto que retornarei em breve.

E finalmente à tabela *ArquivoAutenticado* acrescentamos os campos que a relaciona com outras tabelas: *CodigoContribuinte* e *CodigoEcf* (ver Figura 2) , definindo assim o nosso banco de dados relacional.

Perceba que cada Arquivo Autenticado é marcado - é único - mediante a adição de dois campos - as chaves primárias das outras tabelas.

4. Conclusão

Nesse primeiro tutorial, de posse de apenas um documento, estruturamos um bancos de dados relacional.

Colecionar os documentos com as informações que se quer armazenar é tarefa primária para a formação do banco de dados.

Discutimos o significado de sistemas de gerenciamento de bancos de dados, SQL e as etapas necessárias para se formar um banco de dados.

Em um próximo encontro, entraremos em detalhes conceituais, usando o MS Access no que concerne a inicialização das tabelas, campos, consultas e relatórios, afim de chegarmos ao abjetivo inicial proposto por esse trabalho: Imprimir o Termo de Leitura e Cópia de Dados gravados em Ecf. Acrescentaremos alguns conceitos fundamentais para o entendimento de banco de dados relacionais, quais sejam: chaves primárias, chaves externas, regras de integridade referencial, entidades e outros.

Como observação final, perceba as **convenções** utilizadas nesse trabalho: os nomes das tabelas e campos são substantivos singulares, sem sinais diacríticos, e sempre iniciam com letra maiúscula (Ecf); se a palavra é um substantivo composto (ou adjetivo-substantivado), a letra de junção das duas palavras é capitalizada (ArquivoAutenticado); não separamos as palavras.

Procure nomes significativos; a semântica importa; o nome da tabela, por exemplo, representa cada objeto que nela está armazenado.

O motivo de tais convenções provém de experiências anteriores de estudiosos no assunto; cada artefato (uma tabela, um campo) é tratado como um objeto, daí melhor nomeá-los de maneira consistente, para futuramente, ao necessitarmos citá-los, saberemos seu nome e formato prováveis.

A convenção de nomes é opcional, mas altamente recomendável !

oce pode escolher a convenção que melhor se adeque ao seu estilo de escrita; porém, uma vez escolhida, adote-a permanente e consistentemente:

Importante

A consistência do banco de dados está diretamente relacionado a consistência da nomenclatura adotada pelo seu autor


Obrigado e até o próximo encontro !

5. Recursos

O site da Oficial da Microsoft é bem didático e merece uma visita (<http://office.microsoft.com/pt-br/access/default.aspx>).

Esse outro site, <http://www.databasedev.co.uk/downloads.html>, possui diversos exemplos de bancos de dados.

6. Sobre o autor

Gilberto Jr <giljr.2009@gmail.com> está Auditor Fiscal do Estado de Rondônia, lotado em Ji-Paraná RO; é java-aficcionado e interessado em Information Technology (IT) applied. Made with 

Glossário

Apreender	v.t. Fazer apreensão; tomar, prender: apreender os documentos. Confiscar, apropriar-se judicialmente: apreender os bens. Aprender, compreender: apreender o significado do texto.
Sistema de Gerenciamento de Banco de Dados	<i>Database Managenent system - DBMS</i> - Conjunto de programas de computador que controla a criação, manutenção e o uso de banco da dados pelo computador ou pela organização e seus usuários; sistema de pacote de software que ajuda no uso de uma coleção de informações e arquivos, conhecidos como banco de dados, e permite que aplicativos acessem-nos facilmente.
Driver	<p>No contexto da informática, a função de um driver de dispositivo é aceitar requerimentos abstratos do software independente de dispositivo acima dele e cuidar para que a solicitação seja executada, permitindo que o software interaja com o dispositivo.</p> <p>Um driver de dispositivo simplifica a tarefa da aplicação atuando como um tradutor entre o dispositivo e as aplicações ou o sistema operacional. O código de alto nível das aplicações pode ser escrito independentemente do dispositivo que será utilizado. Qualquer versão de um dispositivo, como uma impressora, requer os seus próprios comandos. Entretanto, a maioria das aplicações acessam aos dispositivos usando comandos genéricos de alto-nível, como println, que imprime um texto. O driver converte esses comandos genéricos e converte para um código de baixo-nível interpretado pelo dispositivo. Em uma melhor tradução, drive do verbo driver, dirige um hardware a um software, ou seja, é o meio de comunicação de um hardware (extraído http://pt.wikipedia.org).</p>
Structured Query Language - SQL	É uma linguagem para bancos de dados desenhada para Sistemas de Gerenciamento Bancos de Dados Relacionais e originalmente baseada em Álgebra Relacional; Seus objetivos incluem consultas de informações (data query), atualização (update), criação de schema, modificações (delete) e acesso a dados (select); SQL foi influenciada pelo trabalho A Relacional Model of Data for Large shared Data Banks, de Edgar F. Codd, e se tornou a mais usada linguagem para bancos de dados relacionais.