

BANCOS DE DADOS

Introdução

Banco de Dados é, na verdade, uma forma de armazenar informações estruturadamente, utilizando-se de campos para melhor dividir e encontrar esses dados. No conceito de Banco de Dados estão inseridas as estruturas que permitem o armazenamento e os processos que localizam e manipulam os dados.

Nome	Endereço	Telefone
Jorge Almeida	Rua dos Pinhais, 37	3241-4242
Marília Azevedo	Rua Demócrito, 234	3221-2211
Antonio Carvalho	Av. Ibiapina, 143	3251-0423
Heitor Carneiro Leão	Rua Rio Tejió, 157	3446-6565

O exemplo anterior mostra uma tabela, com 3 colunas e 5 linhas. Esse é um pequeno modelo de um banco de dados, Nome, Endereço e Telefone são Campos e as linhas de dados são chamadas de Registros.

- **Tabela:** Estrutura bidimensional formada por linhas e colunas, para armazenar os dados. Hoje em dia, um Banco de Dados é formado por várias tabelas.
- **Campo:** Divisão das tabelas, cada campo permite armazenar um tipo específico de informação. Normalmente, os campos são apresentados como as colunas.
- **Registro:** É o nome dado a uma informação completa da tabela. É a linha da tabela.

Campos, como eles podem ser?

Um campo é uma divisão da tabela, portanto, um local para recebimento de um tipo de dado individual.

Um campo aceita apenas um tipo de dado definido em sua estrutura.

- **Número:** armazena apenas valores numéricos, é usado para informações que possam passar por operações aritméticas;
- **Texto:** o campo aceita caracteres alfanuméricos, é necessário definir a quantidade de caracteres que o campo vai aceitar

Os SGBDs (Sistema Gerenciador de Bancos de Dados) são aplicativos (programas) que permitem criar, alterar, e excluir componentes da estrutura dos Bancos de Dados, bem como inserir, alterar e excluir dados nestes.

Os SGBDs mais utilizados hoje em dia é o Access (desenvolvido pela Microsoft), que é muito utilizado para bancos de dados pessoais e menos robustos.

Já o SQL Server e Oracle, são utilizados em projetos com grande volume de dados (grandes empresas).

Os bancos de dados desses programas são conhecidos como Bancos de Dados Relacionais, pois são formados por várias tabelas que se relacionam entre si. Portanto, podemos chamá-los de SGBDR (Sistemas Gerenciadores de Bancos de Dados Relacionais).

Bancos de Dados Relacionais

São bancos de dados formados por várias tabelas que estão relacionadas entre si por alguns campos comuns.

Chave Primária

É um campo de uma tabela que recebe uma "patente" especial. A informação contida neste campo não pode se repetir em dois registros diferentes nem pode ter valor Nulo (vazio).

Cod	Nome	Telefone	Endereço	CPF
1	José Carlos	251-2531	Rua Amilgar, 32	232102325-8
2	Antonio Martins	3369-9898	Av. dos Pilares,	020141478-7
3	Cláudia Simões	3231-0913	Av. João de Bar	525178124-9
4	Eduardo Ferraz	8112-5658	Rua Amarildo Be	123456789-4
*	0			

Registro: 1 de 4

Relacionamentos

Relacionamentos são as ligações entre campos de tabelas diferentes em um BD Relacional. Esses relacionamentos são usados para evitar a redundância de dados (repetição).

Chave Estrangeira

É um campo de uma tabela usado para ligar(relacionar) duas tabelas distintas, evitando a redundância dos dados nessa tabela.

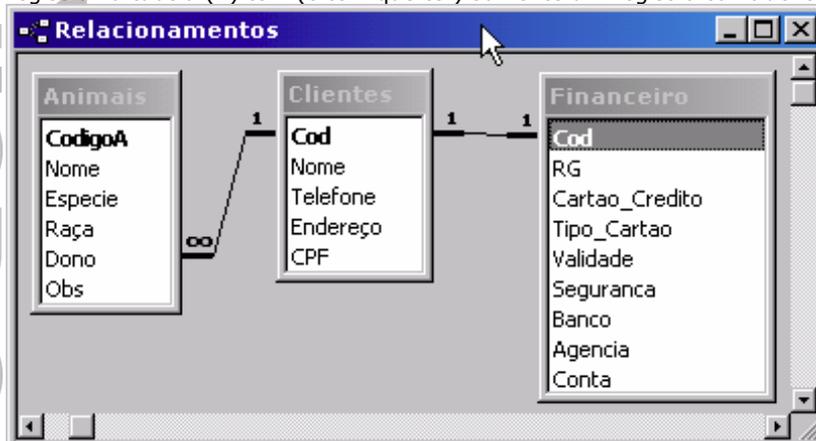
CodC	CodigoA	Data	Resultadc	Medica	Valor	Obs
1	4	12/3/2003	O animal foi col	32	R\$ 70,00	O animal se mo
2	4	18/3/2003	O animal picrol	32	R\$ 70,00	Dose aumentac
3	3	19/3/2003	Tentamos reani		R\$ 120,00	A cirurgia foi a
5	2	21/3/2003	Às custas de r	12	R\$ 70,00	
6	4	23/3/2003	O animal está	32	R\$ 70,00	São Francisco
	0				R\$ 0,00	

Tipos de Relacionamento

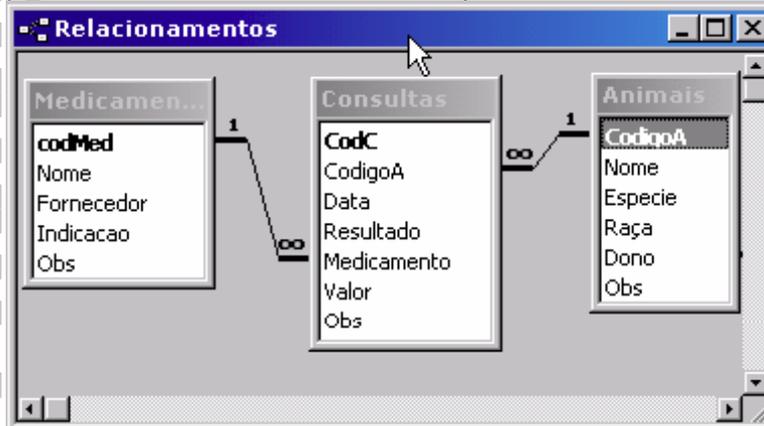
Um-para-muitos: É a relação entre a chave primária de uma tabela (A) e um campo comum em outra tabela (B), ou seja, um registro na tabela (A) pode ter muitos registros coincidentes na tabela (B), mas um registro na tabela (B) tem (e tem que ter) só um registro coincidente na tabela (A).



Um-para-um: Cada registro na tabela (A) pode ter somente um registro coincidente na tabela (B), e cada registro na tabela (B) tem (e tem que ter) somente um registro coincidente na tabela (A)



Muitos-para-muitos: um registro na tabela (A) pode ter muitos registros coincidentes na tabela (B,) e um registro na tabela (B) pode ter muitos registros coincidentes na tabela (A). Esse tipo de relacionamento só é possível definindo-se uma terceira tabela (denominada tabela de união).



SQL - LINGUAGEM DE CONSULTA ESTRUTURADA

É um conjunto de instruções que permitem ao usuário manipular os dados armazenados nas tabelas e a própria estrutura do Banco.

A linguagem de consulta mais usada atualmente, em qualquer que seja o SGBD, é o SQL (Structured Query Language - ou Linguagem de Consulta Estruturada).

FUNÇÕES DO SQL

O SQL possui funções próprias que calculam valores agrupando mais de uma linha. Pode ser a contagem de linhas (registros) ou um cálculo com um campo numérico da tabela.

AVG() : calcula a média de uma coluna de dados numéricos.

Qual é a média de débitos dos alunos da cidade "01"

```
Select avg(debito) from alunos where cidade = "01"
```

COUNT(): conta o número de itens selecionado em uma coluna.

Quantos alunos do sexo feminino estudam o curso "02"

```
select count(*) from alunos where sexo = "F" and curso = "02"
```

MIN() : determina o menor valor de Item_Selecionado em uma coluna.

Qual é o menor e o maior débito dos alunos do turno matutino

```
select min(debito), max(debito) from alunos where turno = "M"
```

MAX() : determina o maior valor de Item_Selecionado em uma coluna.

SUM() : totaliza uma coluna de dados numéricos. Exemplos:

Qual é a soma e quantos alunos devem acima de R\$ 100,00

```
select sum(debito), count(*) from alunos where debito > 100
```

Há dois tipos de instruções no SQL:

- DML (Linguagem de Manipulação de Dados) que permitem inserções, alterações e exclusões nos registros, e as instruções.
- DDL (Linguagem de Definição de Dados) que permitem alterações na estrutura do Banco, como a criação de tabelas e campos, a alteração e a exclusão dos mesmos.

As principais instruções DML da Linguagem SQL são SELECT, INSERT INTO, UPDATE, DELETE.

As principais instruções DDL são CREATE, ALTER e DROP.

INSTRUÇÕES DML

SELECT

Esta instrução seleciona registros e campos específicos de uma tabela.

Sintaxe: SELECT <Campos> FROM <Tabelas> WHERE <Critérios> ORDER BY <Campo> DESC
 SELECT * FROM Animais WHERE Espécie="Canino" ORDER BY Espécie DESC (selecionar todos os campos da tabela Animais da espécie Canino, mostrando os registros em ordem decrescente)

- **SELECT:** seleciona os campos para apresentar os dados. O * (asterisco) indica TODOS os campos.
- **FROM:** tabela de origem dos campos. Para mais de uma tabela, separe por , (vírgula).
 SELECT * FROM alunos
- **WHERE:** indica o critério(filtro) de apresentação dos dados.
 Select * from alunos where cidade in ("01","04","37")
- **GROUP BY:** Agrupa as linhas de consulta com base nos valores de uma ou mais colunas.
 Select curso, count(*) Quantidade From alunos GROUP BY CURSO
- **HAVING:** Especifica uma condição de filtragem que os grupos devem satisfazer para que sejam incluídos nos resultados da consulta. HAVING deve ser utilizada com GROUP BY.
 Select curso, count(*) Quantidade From alunos group by curso HAVING count(*) = 44
- **ORDER BY:** ordena os dados pelo campo citado. DESC, ordem decrescente.
 SELECT aluno, nome,curso FROM Alunos ORDER BY curso, nome

INSERT INTO

Permite inserir dados nas tabelas do BD.

Sintaxe: INSERT INTO <Tabela> (campo1, campo2, campo3) <VALUES> (valor1, valor2, valor3)
 INSERT INTO Animais (CodigoA, Nome, Espécie, Raça, Dono) VALUES (9, "Shazam", "Felino", "Persa", 3) -
 (inserir um novo Animal de código 9, Nome Shazam, espécie Felino, Raça Persa e pertencente ao cliente 3)

UPDATE

Permite que o usuário altere os valores contidos em um ou vários registros.

Sintaxe: UPDATE <Tabela> SET <Campo=Valor> WHERE <Critérios>
 UPDATE Animais SET Espécie="Felino" WHERE Dono=2 - (se todos os animais do Cliente 2 forem, na verdade, gatos)

DELETE

Exclui um ou mais registros de acordo com um critério.

Sintaxe: DELETE * FROM <Tabela> WHERE <Critérios>
 DELETE * FROM Animais WHERE Espécie="Felino" - (excluir todos os gatos da tabela Animais do nosso Banco de Dados)

Lembre-se, a instrução DELETE não apaga apenas um campo do registro (como por exemplo, apagar apenas os nomes dos animais), esta instrução exclui o registro inteiro da tabela.

INSTRUÇÕES DDL

CREATE

Essa instrução cria diversas entidades diferentes num banco de dados, como tabelas e views (visões), entre outros. Para criar uma tabela, usamos a seguinte sintaxe:

CREATE TABLE Tabela (Campo1 Tipo, Campo2 Tipo, etc.)
 Surgirá uma nova tabela no Banco com as configurações citadas na instrução acima.

ALTER

Esta instrução permite alterar a estrutura de uma tabela criada a partir da instrução CREATE TABLE. Com esse comando, podemos criar, modificar e excluir campos das tabelas do banco. A sintaxe básica para adicionar um campo é:

ALTER TABLE Tabela ADD COLUMN Campo Tipo(tamanho)

...modificar um campo é:

ALTER TABLE Tabela ALTER COLUMN Campo Tipo(tamanho)

...excluir um campo é:

ALTER TABLE Tabela DROP COLUMN Campo

Exemplo: para adicionar, à tabela Animais, um campo chamado Sexo, do tipo texto, com um caractere (apenas espaço suficiente para preencher com M ou F), a instrução seria a seguinte:

ALTER TABLE Animais ADD COLUMN Sexo TEXT(1)

DROP

A instrução DROP da linguagem SQL permite que o usuário exclua uma tabela do Banco de Dados. A sintaxe básica deste comando é:

DROP Tabela

Exemplo, se o projetista do banco deseja excluir a tabela de Animais basta acionar o comando: DROP Animais

TRANSACTION – Transação

É um conjunto de operações que é analisado de forma ATÔMICA (ou tudo ou nada)...

Quando uma transferência de fundos é feita entre as contas de fulano e beltrano, muitas operações são realizadas:

INICIO

- a) verifica-se o saldo de fulano
- b) subtrai-se o valor desejado para a transferência do saldo de fulano
- c) subtrai-se o valor da agencia em questão
- d) adiciona-se o valor transferido no saldo de beltrano
- e) adiciona-se o valor na agencia de beltrano

FIM

Por que atômica??? Porque se algo der errado em qualquer um dos passos da transação, NENHUM DOS PASSOS DEVERÁ OCORRER. Já imaginou uma pane depois do passo C???? Fulano com menos dinheiro e beltrano sem receber essa quantia??? INADMISSIVEL!!!!

Toda transação termina ou com um COMMIT (quando tudo deu certo) ou com uma instrução ROLLBACK (quando há um problema)...

START TRANSACTION

- a) verifica-se o saldo de fulano
- b) subtrai-se o valor desejado para a transferência do saldo de fulano SE saldo < 0 então ROLLBACK
- c) subtrai-se o valor da agencia em questão
- d) adiciona-se o valor transferido no saldo de beltrano SE conta inexistente então ROLLBACK
- e) adiciona-se o valor na agencia de beltrano SE agência inexistente então ROLLBACK

COMMIT TRANSACTION

As transações só fazem alterações nos bancos de dados se encontram o COMMIT (ou seja, quando tudo termina bem)...

VISÃO (VIEW)

É uma tabela virtual definida através de consulta. Não há um espaço em disco no qual seus registros estão armazenados.

Exemplo - Visão

```
SELECT * FROM AniverariantesAgosto WHERE nome LIKE 'A%'
```

PROCEDIMENTO ARMAZENADO (STORE PROCEDURE- SP)

São rotinas armazenadas, normalmente, dentro do banco de dados, com funções específicas. São utilizadas quando há a necessidade de se executar um procedimento grande e pesado.

Objetivo

- Assegurar as restrições de integridade
- Melhorar a segurança no acesso aos dados
- Melhorar o desempenho das aplicações
- Reduzir o volume de dados trafegados na rede
- Melhorar a manutenção do software

Exemplo - Stored Procedure

```
CREATE PROCEDURE p_exclui_pessoa
(x_cpf IN CHAR(11)) AS
x_nome VARCHAR(50);
x_end VARCHAR(50);
BEGIN
SELECT nome, endereco INTO x_nome, x_end FROM pessoa
WHERE cpf = x_cpf;
INSERT INTO EX_CLIENTE VALUES
(x_cpf, x_nome, x_end);
DELETE FROM PESSOA WHERE CPF = x_cpf;
EXCEPTION
...
END p_exclui_pessoa;
```

GATILHO (TRIGGER)

Procedimento disparado automaticamente pelo SGBD em resposta a um evento específico do banco de dados. Semelhante a um procedimento armazenado (Stored Procedure), mas tem a sua execução disparada pelo SGBD quando ocorre um evento pré-estabelecido. O evento é associado à tentativa de execução de operações sobre uma tabela (inclusão, exclusão e alteração)

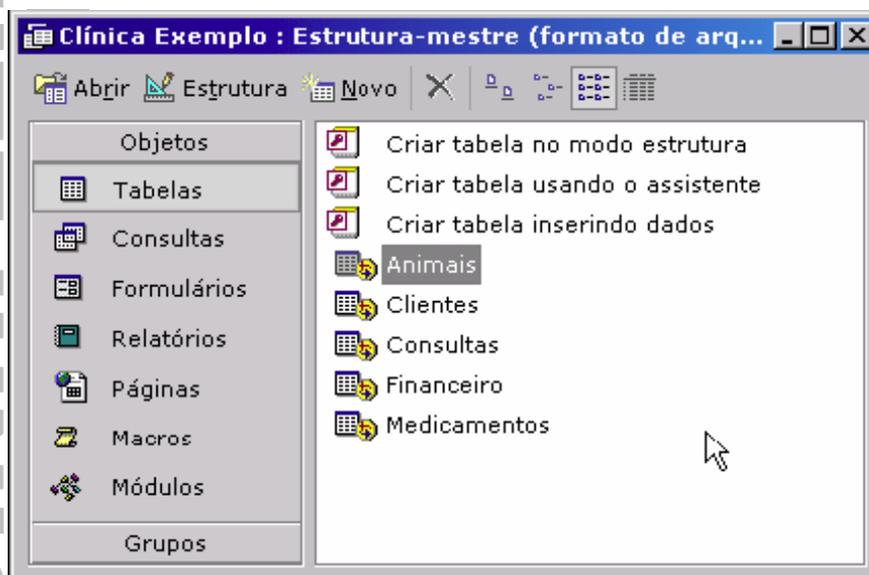
Objetivo

- O mesmo das SP.

```
CREATE TRIGGER t_exclui_pessoa
BEFORE DELETE ON pessoa
FOR EACH ROW
BEGIN
    INSERT INTO EX_CLIENTE VALUES
    (:old.cpf, :old.nome, :old.end);
END t_exclui_pessoa;
```

MICROSOFT ACCESS

É um SGBDR de pequeno porte para ser usado com Bancos de Dados pessoais e de pequenas e médias empresas. O Microsoft Access está inserido no Microsoft Office, a suíte de programas de escritório da Microsoft.



Uma das principais características a respeito do Access é a sua facilidade de uso, tanto para a criação da estrutura do Banco de Dados em si, como para a manipulação de dados por parte dos usuários finais do Banco.

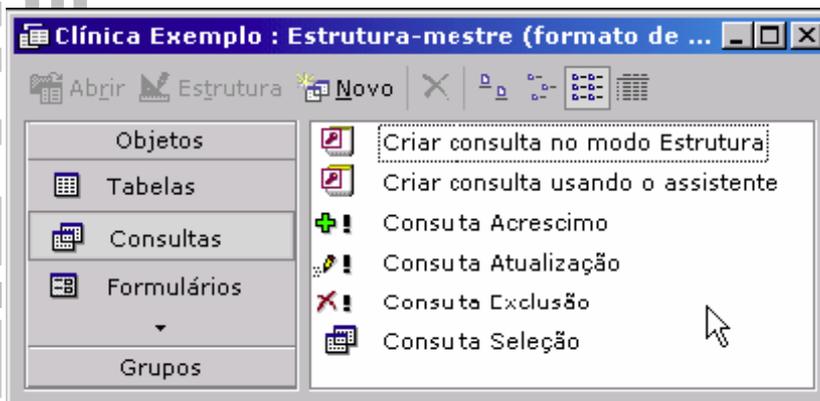
O Microsoft Access permite que sejam criados e modificados vários objetos para o Banco de Dados.

- **Tabelas (Tables):** São as tabelas do banco, feitas para armazenar os dados. As tabelas são os componentes básicos de um BD. Sem tabelas não é possível criar qualquer um dos outros componentes.
- **Consultas (Queries):** São os objetos que buscam e alteram dados das tabelas. Na verdade, as consultas são apenas instruções SQL gravadas para uso posterior. A forma de criar as consultas é bastante simples, tornando desnecessário que o usuário conheça a linguagem SQL.
- **Formulários (Forms):** São as "janelas" que o projetista cria para fazer interface com o usuário que irá utilizar o Banco. Os formulários são uma espécie de "maquiagem" para as tabelas e consultas aparecem mais "agradáveis" na tela.
- **Relatórios (Reports):** São os documentos que serão impressos no Banco de Dados. Pode-se construir Relatórios para tabelas ou para Consultas.
- **Macros:** São conjuntos de ações escritas em sequência para que o Access realize operações de forma automática. As macros no Access não têm o mesmo "formato" que no Excel ou Word.
- **Módulos (Modules):** São programas criados em VBA (Visual Basic for Applications), a linguagem de programação que acompanha os aplicativos do Microsoft Office. Os módulos estão para o Access assim como as Macros estão para o Word e Excel.
- **Páginas (pages):** Este recurso acompanha apenas as mais recentes versões do Microsoft Access e permite que o programa crie páginas dinâmicas para a Web que acessem os dados do Banco. Essas páginas serão construídas nas linguagens ASP +HTML.

Tipos de Consultas do Access

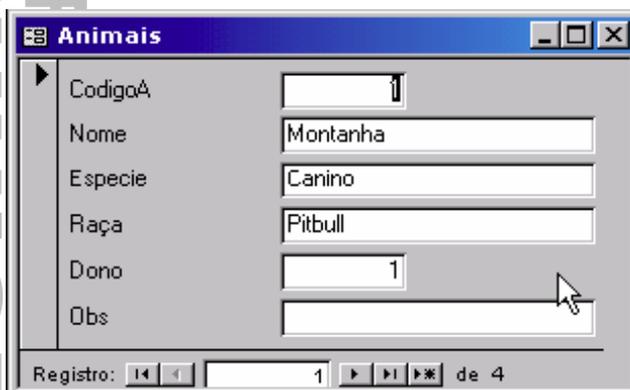
Uma consulta, para o Access, é tão somente uma expressão SQL armazenada para uso posterior.

- **Consulta Seleção:** É uma consulta construída a partir da instrução SELECT. Esta consulta serve para selecionar campos e registros definidos das tabelas do Banco;
- **Consulta Acréscimo:** É uma consulta feita a partir da instrução INSERT INTO. Esta consulta é utilizada para adicionar registros às tabelas;
- **Consulta Exclusão:** É uma consulta baseada na instrução DELETE. É utilizada para excluir registros de uma tabela;
- **Consulta Atualização:** É usada para alterar os valores de determinados campos em uma tabela. Essa consulta é construída por intermédio da instrução UPDATE;
- **Consulta Criar Tabela:** Uma consulta que permite a criação de uma tabela no Banco de Dados. Nessa consulta, é utilizada a instrução CREATE TABLE.

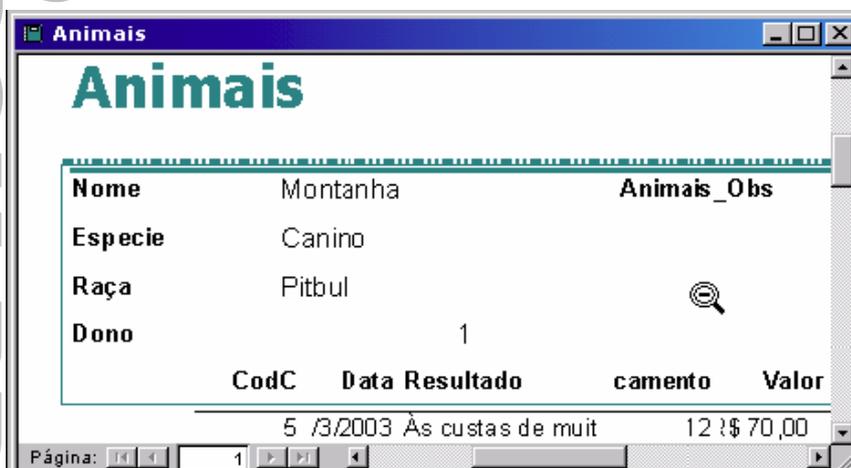


Formulários e Relatórios no Access

Formulários, no Access, são as janelas que o projetista cria para servir de interface entre o usuário e os componentes básicos do Banco (tabelas e consultas). As telas usadas pelos atendentes para preencher os dados dos cadastros dos clientes são os formulários.



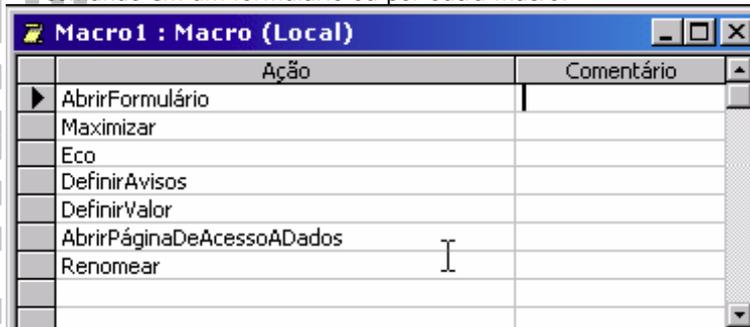
Relatórios são os documentos impressos que apresentarão dados das tabelas e consultas do Banco. Se um banco de dado tem que emitir uma "nota" ou "recibo" da consulta veterinária, esse documento será um relatório.



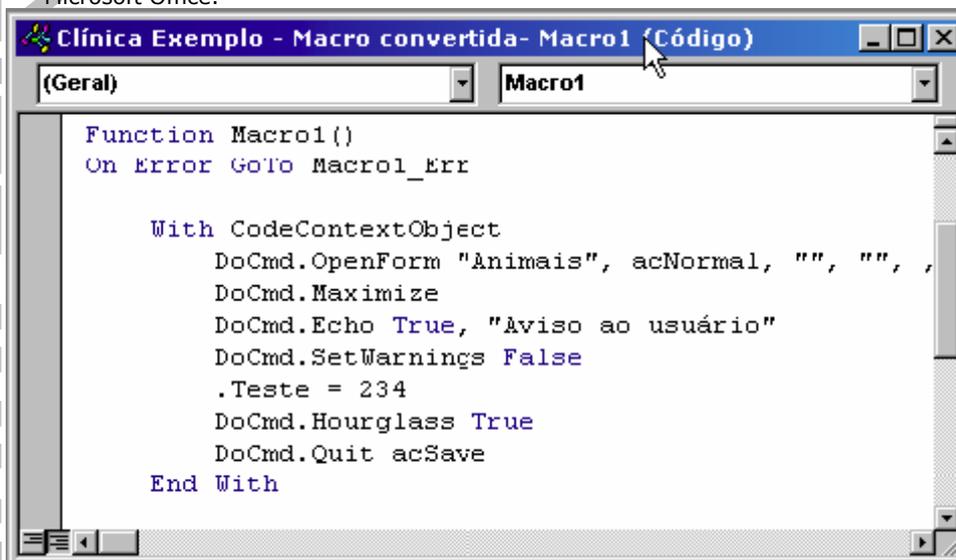
Macros e Módulos

Para automatizar tarefas no Access, podemos usar dois tipos de objetos, as Macros e os Módulos.

- **Macros:** são seqüências de ações predefinidas pelo usuário usadas basicamente para realizar tarefas simples, que já estão definidas numa lista no Access. Uma macro pode ser acionada por um botão de comando em um formulário ou por outra macro.



- **Módulo:** o usuário normalmente precisa conhecer a linguagem de programação VBA, que acompanha o Microsoft Office.



BANCOS DE DADOS NAS EMPRESAS

Data Warehouse: Bancos de dados enormes, que armazenam informações de todos os setores de uma empresa para dar suporte à tomada de decisões estratégicas. Exemplo: Os sistemas que armazenam dados de clientes, fornecedores, produtos, entradas, saídas, perdas, pessoal, contabilidade de uma empresa (todos juntos em um único banco).

Data Mart: Bancos de dados que armazenam informações normalmente de um determinado setor da empresa. Serve, como o DW, para auxiliar na tomada de decisões vitais para a empresa, mas não resume dados de várias fontes ou setores.

Data Mining: Algo como "Mineração de Dados", definem as ações realizadas, no Banco de Dados, para detectar relacionamentos entre as informações armazenadas. Um exemplo: em redes varejistas, programas que identificam clientes com interesses comuns.

OLAP: Significa "Processo Analítico On-line" e descreve uma ferramenta, usada pelos usuários finais, capaz de extrair dados de um Data Warehouse e analisá-los. A consulta ao DW é interativa, permitindo aos usuários verificarem o "porquê" dos resultados obtidos.

Business Intelligence: Ou "Inteligência de Negócios" é dos conceitos mais "na crista da onda" hoje em dia no que concerne a Gestão Empresarial. A inteligência resulta da coleta de dados, transformação desses dados em informação útil e, por fim a análise dessas informações contextualizadas para que sejam aplicadas a processos decisivos e gerem vantagens competitivas. Em suma, BI descreve uma série de componentes, técnicas e recursos para o armazenamento de dados e a sua utilização em prol da estratégia de negócios.

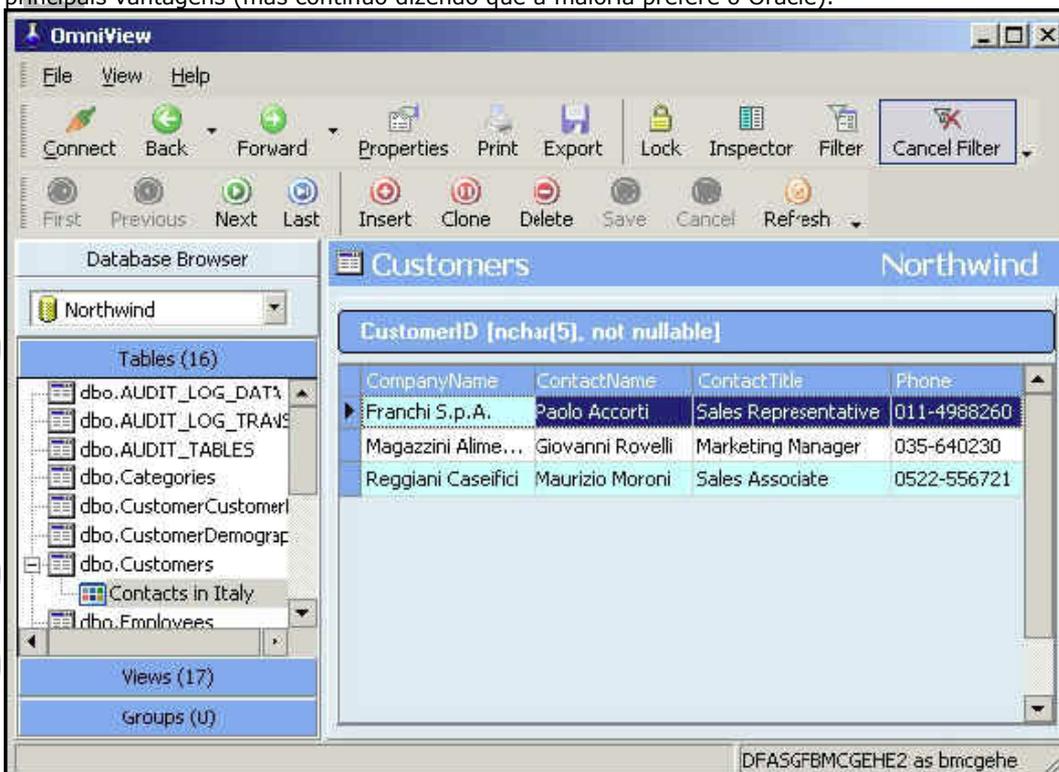
SGBD'S CORPORATIVOS

O Access é um excelente Gerenciador de Bancos de Dados, mas é indicado para aplicações de pequeno porte. Para o gerenciamento de grandes volumes de dados, recomenda-se a utilização de um SGBDR mais robusto. Os dois nomes mais conhecidos são o Oracle, da Oracle Corp., SQL Server da Microsoft e o MySQL (SGBD Gratuito para ambientes Linux e Windows).

O Oracle é o SGBDR preferido pela maioria das empresas, sua licença de uso é muito cara, mas ele está presente em grande parte dos sistemas de armazenamento de grandes volumes de dados.



O SQL Server, da Microsoft é o principal concorrente do Oracle. Sua facilidade de uso e configuração são suas principais vantagens (mas continuo dizendo que a maioria prefere o Oracle).



O MySQL, um SGBDR completamente gratuito para ser usado em ambientes LINUX (há versões para Windows também).

QUESTÕES DE CONCURSO

1) Acerca dos conceitos de Bancos de Dados, julgue os itens a seguir:

- I. Uma tabela com mais de 10.000 registros ordenados precisa ter, em cada campo, uma chave primária para evitar erros de integridade dos dados.
- II. É possível alterar valores nos dados de uma tabela que não possui chave primária através de instruções SQL.
- III. Um banco de dados relacional tem, associada ao seu conceito básico, a existência de várias tabelas que possuem campos interligados por estruturas conhecidas como relacionamentos.
- IV. Não pode haver, em uma tabela com um campo chave, registros que apresentem o valor desse campo sendo 0 (zero) ou registros com esse campo duplicado (igual ao de outro registro).

Estão corretos os itens:

- a) I e II b) III e IV c) II e IV d) II e III e) I e III

2) Analise a tabela abaixo:

Nome	E-Mail	Mensalidade	Saldo Devedor
Jorge Rufino	jorge@provedor	30,00	120,00
Anderson Silva	anderson@trt	30,00	150,00
Marcus Macedo	marcusm@faculdade	30,00	190,00
Anita Santos	anita@tv	30,00	195,00

Sabendo que os dados acima citados foram extraídos de uma tabela chamada clientes, cujos campos são nomecli, mailcli, mensalcli, telefonecli, cpfcli e saldocli, e que possui mais de 200 registros, assinale alternativa que indica a instrução SQL possivelmente usada para apresentar o resultado mostrado na tabela acima:

- a) SELECT * FROM clientes WHERE mensalcli=30 AND saldocli>120
- b) SELECT nomecli AS Nome, mailcli AS E-Mail, mensalcli AS Mensalidade, saldocli AS "Saldo Devedor" FROM clientes WHERE mensalcli=30 AND saldocli>=120 ORDER BY saldocli
- c) SELECT nomecli AS Nome, mailcli AS E-Mail, mensalcli AS Mensalidade, saldocli AS "Saldo Devedor" FROM clientes WHERE saldocli<120 ORDER BY nomecli
- d) SELECT nomecli AS Nome, mailcli AS E-Mail, mensalcli AS Mensalidade, saldocli AS "Saldo Devedor" FROM clientes WHERE mensalcli<>30
- e) SELECT nomecli, mailcli, mensalcli, saldocli FROM clientes WHERE mensalcli=30 AND saldocli>120 ORDER BY saldocli DESC

3) Ainda com relação à tabela mostrada acima, é correto afirmar que o registro apresentado na segunda linha dos dados da tabela (Cliente Anderson) seria apresentado na primeira linha se a cláusula ORDER BY, do comando descrito pela questão anterior, fosse trocado por:

- a) ORDER BY saldocli DESC
- b) ORDER BY mensalcli
- c) ORDER BY nomecli DESC
- d) ORDER BY nomecli
- e) ORDER BY saldocli

4) Relacionamentos são ligações entre campos de tabelas diferentes para evitar duplicidade de dados e manter a integridade deles. Quanto aos vários tipos de relacionamentos, podemos afirmar que:

- a) Um relacionamento um-para-muitos se comporta de forma idêntica a um relacionamento um-para-um em tabelas que não possuem chaves primárias em todos os registros.
- b) Um relacionamento um-para-um entre as tabelas A e B permite a inserção de dados duplicados na tabela B, mas não na tabela A.
- c) Um relacionamento um-para-muitos entre um campo da tabela A e um campo da tabela B permite a duplicação do valor do referido campo na tabela B. Não é, porém, permitida a duplicação de dados no campo da tabela A.
- d) Um relacionamento muitos-para-muitos é construído diretamente entre duas tabelas que não possuem chave primária.
- e) Um relacionamento um-para-muitos entre as tabelas A e B só é construído mediante a presença de uma tabela C intermediária.

5) O uso da informática como uma ferramenta de auxílio às tomadas de decisão nas empresas através da diminuição do tempo de análise dos dados gerou uma série de conceitos e termos novos, como Business Intelligence, Data Mart, Data Warehouse, OLAP, Data Mining. Um Data Warehouse, por exemplo, pode ser definido como sendo:

- a) Um banco de dados corporativo que armazena centralizadamente os dados de vários departamentos da empresa.
- b) Um processo de armazenamento de dados usado para controle de estoque em armazéns e fábricas.
- c) Um processo de seleção dos dados pertinentes ao nível estratégico da corporação.
- d) Um banco de dados departamental, que guarda informações relevantes a um determinado setor da empresa.
- e) Um sistema de auxílio à tomada de decisão por parte dos usuários operacionais da empresa.

GABARITO

1) Letra D

I. FALSO: Chave primária é a marcação que UM dos campos recebe. Um campo com essa característica não pode ter valor nulo (vazio) ou ter valores repetidos.

II. VERDADEIRO

III. VERDADEIRO

IV. FALSO: Valor 0 (zero) não é nulo. Então, pode haver um registro com o campo chave preenchido pelo valor 0 (zero) sem problemas!

2) Letra B

3) Letra D

4) Letra C

a) FALSO: 1-n e 1-1 não são iguais de jeito nenhum!

b) FALSO: em relacionamentos 1-1, os valores nos campos relacionados só aparecem uma vez em cada tabela

c) VERDADEIRO: no relacionamento 1-n, a tabela A apresenta o valor do campo relacionado apenas uma vez, mas na tabela B o valor pode aparecer várias vezes!

d) FALSO: é necessária uma tabela intermediária para criar um relacionamento n-n.

e) FALSO: quem precisa de tabela C é o n-n, apenas. Os demais são relacionamentos diretos entre duas tabelas.

5) Letra A

a) OK!

b) Isso é loucura!

c) Isso é DATA MINING.

d) Isso é DATA MART

e) SAD (Sistema de Apoio à Decisão), mas não é o nível operacional que usa, e sim o nível estratégico ou executivo da empresa.