

Banco de Dados

Todos nós sabemos existirem gigantescas bases de dados gerenciando nossas vidas. De fato sabemos que nossa conta bancária faz parte de uma coleção imensa de contas bancárias de nosso banco. Nosso Título Eleitoral ou nosso Cadastro de Pessoa Física, certamente estão armazenados em Bancos de Dados colossais. Sabemos também que quando sacamos dinheiro no Caixa Eletrônico de nosso banco, nosso saldo e as movimentações existentes em nossa conta bancária já estão à nossa disposição.

Nestas situações sabemos que existe uma necessidade em se realizar o armazenamento de uma série de informações que não se encontram efetivamente isoladas umas das outras, ou seja, existe uma ampla gama de dados que se referem a relacionamentos existentes entre as informações a serem manipuladas.

Estes Bancos de Dados, além de manterem todo este volume de dados organizado, também devem permitir atualizações, inclusões e exclusões do volume de dados, sem nunca perder a consistência. E não podemos esquecer que na maioria das vezes estaremos lidando com acessos concorrentes a várias tabelas de nosso banco de dados, algumas vezes com mais de um acesso ao mesmo registro de uma mesma tabela!

O fato de montarmos uma Mala Direta em um micro PC-XT com um drive já faz de nós um autor de um Banco de Dados?

Claro que não! Um Banco de Dados é antes de mais nada uma coleção logicamente coerente de dados com determinada significação intrínseca. Em outras palavras um arquivo contendo uma série de dados de um cliente, um arquivo com dados aleatoriamente gerados e dois arquivos padrão dbf (dBase) que tem uma relação definida entre ambos, não pode ser considerada uma Base de Dados Real.

Um Banco de Dados contém os dados dispostos numa ordem pré-determinada em função de um projeto de sistema, sempre para um propósito muito bem definido.

Um Banco de Dados representará sempre aspectos do Mundo Real. Assim sendo uma Base de Dados (ou Banco de Dados, ou ainda BD) é uma fonte de onde poderemos extrair uma vasta gama de informações derivadas, que possui um nível de interação com eventos como o Mundo Real que representa. A forma mais comum de interação Usuário e Banco de Dados, dá-se através de sistemas específicos que por sua vez acessam o volume de informações geralmente através da linguagem SQL.

Os Administradores de Banco de Dados (DBA) são responsáveis pelo controle ao acesso aos dados e pela coordenação da utilização do BD. Já os projetistas de Banco de Dados (DBP) são analistas que identificam os dados a serem armazenados em um Banco de Dados e pela forma como estes serão representados.

Os Analistas e Programadores de Desenvolvimento, criam sistemas que acessam os dados da forma necessária ao Usuário Final, que é aquele que interage diretamente com o Banco de Dados.

BANCO DE DADOS

Em um sistema de informação, fragmentos de dados são organizados em uma hierarquia. O menor fragmento de dado é o bit. O byte é um grupo de bits que forma um caracter. Um campo é um grupo de caracteres que forma uma palavra, ou um número. Um registro é um conjunto de campos relacionados; um arquivo é um grupo de registros relacionados. O maior elemento da hierarquia, o banco de dados, consiste em arquivos relacionados.

Conceito

Banco de Dados: é uma coleção de dados interrelacionados, representando informações sobre um domínio específico. Basicamente é uma tabela composta de várias linhas divididas em colunas que são identificadas por campos, e cada linha representa um registro de banco de dados. Com esta organização o Excel poderá oferecer recursos de organização e pesquisa extremamente fáceis, independente da quantidade de linhas contidas na base de dados.

Pode ser definido também, como uma coleção de dados organizados de tal forma que possam ser acessados e utilizados por muitas aplicações diferentes. Exemplos: lista telefônica, controle do acervo de uma biblioteca, sistema de controle dos recursos humanos de uma empresa.

Propriedades de um Banco de Dados

A tecnologia aplicada aos métodos de armazenamento de informações vem crescendo e gerando um impacto cada vez maior no uso de computadores, em qualquer área em que os mesmos podem ser aplicados.

Um "banco de dados" pode ser definido como um conjunto de "dados" devidamente relacionados. Por "dados" podemos compreender como "fatos conhecidos" que podem ser armazenados e que possuem um significado implícito. Porém, o significado do termo "banco de dados" é mais restrito que simplesmente a definição dada acima. Um banco de dados possui as seguintes propriedades:

- é uma coleção lógica coerente de dados com um significado inerente;
- uma disposição desordenada dos dados não pode ser referenciada como um banco de dados;
- um banco de dados é projetado, construído e populado com dados para um propósito específico;
- um banco de dados possui um conjunto pré definido de usuários e aplicações;
- um banco de dados representa algum aspecto do mundo real, o qual é chamado de "mini-mundo"; qualquer alteração efetuada no mini-mundo é automaticamente refletida no banco de dados.
- Um banco de dados pode ser criado e mantido por um conjunto de aplicações desenvolvidas especialmente para esta tarefa ou por um "Sistema Gerenciador de Banco de Dados" (SGBD).

Componentes de um Banco de Dados

Um Banco de Dados é composto pelas seguintes partes:

Gerenciador de Acesso ao Disco: O SGBD utiliza o Sistema Operacional para acessar os dados armazenados em disco, controlando o acesso concorrente às tabelas do Banco de Dados. O Gerenciador controla todas as pesquisas (queries) solicitadas pelos usuários no modo interativo, os acessos do compilador DML, os acessos feitos pelo Processador do Banco de Dados ao Dicionário de Dados e também aos próprios dados.

O Compilador DDL (Data Definition Language) processa as definições do esquema do Banco de Dados, acessando quando necessário o Dicionário de Dados do Banco de Dados.

O Dicionário de Dados contém o esquema do Banco de Dados, suas tabelas, índices, forma de acesso e relacionamentos existentes.

O Processador do Banco de Dados manipula requisições à própria Base de Dados em tempo

de execução. É o responsável pelas atualizações e integridade da Base de Dados.

O Processador de Pesquisas (queries) dos usuários, analisa as solicitações, e se estas forem consistentes, aciona o Processador do Banco de Dados para acesso efetivo aos dados.

As aplicações fazem seus acessos ao pré-compilador DML da linguagem hospedeira, que os envia ao Compilador DML (Data Manipulation Language) onde são gerados os códigos de acesso ao Banco de Dados.

Banco de dados nas Empresas :

Em todos os sistemas de informação, os dados devem ser organizados e estruturados para que possam ser usados com eficácia. A má organização de arquivos impede que algumas empresas possam acessar grande parte das informações que mantêm.

Porém, quando adequadamente documentado, o dicionário de dados é uma importante ferramenta de resolução de problemas. Ele identifica para os usuários finais e para os especialistas empresariais quais dados existem no banco de dados, sua estrutura, e formato e sua utilização na empresa.

Atualmente, existe uma tendência de mercado em se dizer que qualquer problema será resolvido, caso a empresa adquira um Banco de Dados. Naturalmente, em um ambiente com acesso constante ao Banco de Dados (acesso concorrente, obviamente), onde a segurança seja de vital importância e que o desempenho da aplicação escrita estiver comprometendo a empresa, considerando-se logicamente uma aplicação bem escrita, sem dúvida a aquisição de um Banco de Dados *poderá ser* o primeiro passo na solução do problema.

Analogamente ao que ocorreu com o aparecimento das primeiras linguagens de programação voltadas ao Windows, onde estas foram apresentadas como capazes de alavancar os negócios da empresa, e no geral causaram mais frustração do que solução, a aquisição do Banco de Dados, pode gerar o mesmo tipo de problema.

É fundamental que a empresa candidata a utilizar um Banco de Dados, normatize-se totalmente, pois soluções "quebra-galho", típicas do ambiente que dispõe de um Gerenciador de Arquivo, tendem a ser impossíveis em um ambiente estruturado sobre o Banco de Dados. Portanto, sob pena de se realizar um grande investimento, e não se colher fruto algum, é muito conveniente, que a empresa *antes* de adquirir um Banco de Dados, passe por um processo de adaptação, preferencialmente contando com pessoal especializado, geralmente consultores, que *não* tenham qualquer ligação com fabricantes de Bancos de Dados.

Sistema de Gerenciamento de Bancos de Dados (SGBD)

É um software com recursos específicos para facilitar a manipulação das informações dos bancos de dados e o desenvolvimento de programas aplicativos.

Exemplos: Oracle, Ingres, Paradox, Access, dBase.

Este Sistema envolve quatro componentes principais:

- dados,
- hardware,
- software e

- usuários.

O sistema de bancos de dados pode ser considerado como uma sala de arquivos eletrônica. Existe uma série de métodos, técnicas e ferramentas que visam sistematizar o desenvolvimento de sistemas de bancos de dados.

Para manipulação de um SGBD deve-se considerar algumas regras básicas e claras. Fica implícito que se ao menos uma das características abaixo não estiver presente no nosso "candidato" a SGBD, este poderá ser um GA (Gerenciador de Arquivo) de altíssima qualidade, "quase" um SGBD, mas não um SGBD.

Normas:

Regra 1: Auto-Contenção- Um SGBD não contém apenas os dados em si, mas armazena completamente toda a descrição dos dados, seus relacionamentos e formas de acesso. Normalmente esta regra é chamada de Meta-Base de Dados. Em um GA, em algum momento ao menos, os programas aplicativos declaram estruturas (algo que ocorre tipicamente em C, COBOL e BASIC), ou geram os relacionamentos entre os arquivos (típicos do ambiente xBase). Por exemplo, quando você é obrigado a definir a forma do registro em seu programa, você não está lidando com um SGBD.

Regra 2: Independência dos Dados- Quando as aplicações estiverem realmente imunes a mudanças na estrutura de armazenamento ou na estratégia de acesso aos dados, podemos dizer que esta regra foi atingida. Portanto, nenhuma definição dos dados deverá estar contida nos programas da aplicação. Quando você resolve criar uma nova forma de acesso, um novo índice, *se precisar alterar o código de seu aplicativo*, você não está lidando com um SGBD.

Regra 3: Abstração dos Dados- Em um SGBD real é fornecida ao usuário somente uma representação conceitual dos dados, o que não inclui maiores detalhes sobre sua forma de armazenamento real. O chamado Modelo de Dados é um tipo de abstração utilizada para fornecer esta representação conceitual. Neste modelo, um esquema das tabelas, seus relacionamentos e suas chaves de acesso são exibidas ao usuário, porém nada é afirmado sobre a criação dos índices, ou como serão mantidos, ou qual a relação existente entre as tabelas que deverá ser mantida íntegra. Assim se você desejar inserir um pedido em um cliente inexistente e esta entrada não for automaticamente rejeitada, você não está lidando com um SGBD.

Regra 4: Visões- Um SGBD deve permitir que cada usuário visualize os dados de forma diferente daquela existente previamente no Banco de Dados. Uma visão consiste de um subconjunto de dados do Banco de Dados, necessariamente derivados dos existentes no Banco de Dados, porém estes não deverão estar explicitamente armazenados. Portanto, toda vez que você é obrigado a replicar uma estrutura, para fins de acesso de forma diferenciada por outros aplicativos, você não está lidando com um SGBD.

Regra 5: Transações- Um SGBD deve gerenciar completamente a integridade referencial definida em seu esquema, sem precisar em tempo algum, do auxílio do programa aplicativo. Desta forma exige-se que o banco de dados tenha ao menos uma instrução que permita a gravação de uma série de modificações simultâneas e uma instrução capaz de cancelar um série de modificações. Por exemplo, imaginemos que estejamos cadastrando um pedido para um cliente, que este deseje reservar 5 itens de nosso estoque, que estão disponíveis e portanto são reservados, porém existe um bloqueio financeiro (duplicatas em atraso) que impede a venda. A transação deverá ser desfeita com apenas uma instrução ao Banco de Dados, sem qualquer modificação suplementares nos dados. Caso você se obrigue a corrigir as reservas, através de acessos complementares, você não está lidando com um SGBD.

Regra 6: Acesso Automático- Em um GA uma situação típica é o chamado Dead-Lock, o abraço mortal. Esta situação indesejável pode ocorrer toda vez que um usuário travou um registro em uma tabela e seu próximo passo será travar um registro em uma tabela relacionada à primeira,

porém se este registro estiver previamente travado por outro usuário, o primeiro usuário ficará paralisado, pois, estará esperando o segundo usuário liberar o registro em uso, para que então possa travá-lo e prosseguir sua tarefa. Se por hipótese o segundo usuário necessitar travar o registro travado pelo primeiro usuário (!), afirmamos que ocorreu um abraço mortal, pois cada usuário travou um registro e precisa travar um outro, justamente o registro anteriormente travado pelo outro! Imaginemos um caso onde o responsável pelos pedidos acabou de travar o Registro Item de Pedido, e, necessita travar um registro no Cadastro de Produtos, para indicar uma nova reserva. Se concomitantemente estiver sendo realizada uma tarefa de atualização de pendências na Tabela de Itens, e para tanto, previamente este segundo usuário travou a Tabela de Produtos, temos a ocorrência do abraço mortal. Se a responsabilidade de evitar esta ocorrência for responsabilidade da aplicação, você não está lidando com um SGBD.

Características Gerais de um SGBD

Os SGBD tem sete características operacionais elementares sempre observadas, que passaremos a listar:

Característica 1: Controle de Redundâncias- A redundância consiste no armazenamento de uma mesma informação em locais diferentes, provocando inconsistências. Em um Banco de Dados as informações só se encontram armazenadas em um único local, não existindo duplicação descontrolada dos dados. Quando existem replicações dos dados, estas são decorrentes do processo de armazenagem típica do ambiente Cliente-Servidor, totalmente sob controle do Banco de Dados.

Característica 2: Compartilhamento dos Dados- O SGBD deve incluir software de controle de concorrência ao acesso dos dados, garantindo em qualquer tipo de situação a escrita/leitura de dados sem erros.

Característica 3: Controle de Acesso- O SGBD deve dispor de recursos que possibilitem selecionar a autoridade de cada usuário. Assim um usuário poderá realizar qualquer tipo de acesso, outros poderão ler alguns dados e atualizar outros e outros ainda poderão somente acessar um conjunto restrito de dados para escrita e leitura.

Característica 4: Interfaceamento- Um Banco de Dados deverá disponibilizar formas de acesso gráfico, em linguagem natural, em SQL ou ainda via menus de acesso, não sendo uma "caixa-preta" somente sendo passível de ser acessada por aplicações.

Característica 5: Esquematização- Um Banco de Dados deverá fornecer mecanismos que possibilitem a compreensão do relacionamento existentes entre as tabelas e de sua eventual manutenção.

Característica 6: Controle de Integridade - Um Banco de Dados deverá impedir que aplicações ou acessos pelas interfaces possam comprometer a integridade dos dados.

Característica 7: Backups- O SGBD deverá apresentar facilidade para recuperar falhas de hardware e software, através da existência de arquivos de "pré-imagem" ou de outros recursos automáticos, exigindo minimamente a intervenção de pessoal técnico.

Existe a possibilidade de encontramos Bancos de Dados que não satisfaçam completamente todas as características acima, o que não o invalida como Banco de Dados. Na prática podemos encontrar situações onde a primeira característica não seja importante, pois podemos ter o Banco de Dados baseado totalmente em um único servidor, e as redundâncias podem ser aceitas em

algumas situações sob controle da aplicação (algo não muito recomendado, mas passível de aceitação, em situações onde a existência do nome do cliente em um arquivo contendo duplicatas emitidas, possibilita o acesso a apenas uma tabela sem relacionamentos, e sabe-se de antemão que uma duplicata depois de emitida, não pode ter seu cliente alterado).

A segunda característica (Compartilhamento dos Dados) pode ser desconsiderada principalmente em ambiente de desenvolvimento, ou ainda em aplicações remotas.

O Controle de Acesso pode ser descartado em pequenas empresas, sendo que o aplicativo em questão, mais o software de rede, pode facilmente se imcubir desta característica, no caso de pequenas empresas, com reduzido número de pessoas na área operacional.

O Interfaceamento e a Esquematização, são características sempre disponíveis, o que varia neste caso é qualidade destes componentes, que vai desde o sofrível até o estado da arte. É muito conveniente que esta característica seja muito boa em um Banco de Dados, onde estiverem em atuação mais de um Administrador de Banco de Dados e tivermos um número relativamente alto de sistemas desenvolvidos ou em desenvolvimento neste ambiente.

De fato, quanto maior o número de pessoas envolvidas no desenvolvimento de aplicações e gerenciamento do Banco de Dados, mais importante tornam-se estas duas características, pois cada novo sistema desenvolvido precisará sempre estar adequado ao Banco de Dados da Empresa e aderente aos padrões de acesso utilizados nos sistemas concorrentes.

As interfaces ISQL e WinSQL devem deixar muito claro ao estudante como uma interface pobre (no caso a existente no ISQL) perde muito, quando comparada a uma interface mais recursiva. A esquematização existente no Banco de Dados é muito melhor do que aquela mantida em alguma pasta, em algum arquivo do CPD, que sempre está "um pouquinho" desatualizada.

O Controle de Integridade, é outra característica sempre presente nos Bancos de Dados, mas existem diferenças quando da implementação desta característica. Assim, é comum encontrarmos Bancos de Dados que suportam determinado acesso, enquanto outros não dispõem de recurso equivalente.

O Backup em tempo de execução, é outra característica sempre disponível, porém temos aplicações que invariavelmente são comprometidas por falhas de hardware, e outras, que o mesmo tipo de falha não causa perda alguma de dados ou de integridade. Novamente, cada Banco de Dados tem esta característica melhor ou pior implementada, cabendo ao Administrador de Banco de Dados escolher aquele que lhe oferecer mais segurança.

Devemos ressaltar ainda, que podemos ter um Banco de Dados Modelo A, que respeite integralmente as regras básicas e disponha de todas as características apresentadas, enquanto um Modelo B que apesar de respeitar as regras básicas, não suporte uma ou outra característica desejável, mas tenha um desempenho excelente, enquanto o Modelo A seja apenas razoável no quesito desempenho, nos levará seguramente a escolher o Modelo B como sendo o ganhador para nossa instalação!

Isto ocorre pois, na prática, todo usuário deseja um tempo de resposta muito pequeno. O chamado "prazo de entrega" muito comum em Bancos de Dados operando nos limites de sua capacidade, ou nos casos onde o hardware está muito desatualizado, é fonte de inúmeros problemas para o pessoal de informática. Neste caso é melhor abrimos mão de uma Interface Amigável, de um Gerencialmente Automático de Backups ou ainda de outras características que não julgarmos fundamentais, para nos livrarmos do problema típico de ambiente extremamente comprometido, por má performance do Banco de Dados.

A escolha do Banco de Dados da empresa, portanto é uma decisão muito delicada, na medida

em que está ir acarretar troca de aplicativos e troca de hardware. Os investimentos diretamente aplicados no Banco de Dados, costumam ser infinitamente menores do que aqueles a serem aplicados na empresa, visando sua perfeita adequao ao novo SGBD. Esta deciso, sempre que possvel, deve ser tomada por especialistas em Banco de Dados, com profundos conhecimentos de Anlise de Sistemas, de Banco de Dados e de Software de Gerenciamento de Bases de Dados, de forma a evitar que a empresa escolha um Banco de Dados inadequado aos seus propsitos, e que pouco tempo depois, seja obrigada a perder todos investimento realizado em Software e Hardware.

Objetivos de um Sistema de Bancos de Dados

- Isolar os usurios dos detalhes mais internos do banco de dados (abstrao de dados).
- Prover independncia de dados s aplicaes (estrutura fsica de armazenamento e  estratgia de acesso).

Vantagens:

- rapidez na manipulao e no acesso  informao,
- reduo do esforo humano (desenvolvimento e utilizao),
- disponibilizao da informao no tempo necessrio,
- controle integrado de informaes distribudas fisicamente,
- reduo de redundncia e de inconsistncia de informaes,
- compartilhamento de dados,
- aplicao automtica de restries de segurana,
- reduo de problemas de integridade.

O sistema de bancos de dados deve prover uma viso abstrata de dados para os usurios.

Nveis de Abstrao

Nvel fsico: nvel mais baixo de abstrao. Descreve como os dados esto realmente armazenados, englobando estruturas complexas de baixo nvel.

Nvel conceitual: descreve quais dados esto armazenados e seus relacionamentos. Neste nvel, o banco de dados  descrito atravs de estruturas relativamente simples, que podem envolver estruturas complexas no nvel fsico.

Nvel de vises do usurio: descreve partes do banco de dados, de acordo com as necessidades de cada usurio, individualmente. Conjunto de ferramentas conceituais para a descrio dos dados, dos relacionamentos entre os mesmos e das restries de consistncia e integridade.

Dividem-se em:

- baseados em objetos,
- baseados em registros.

Modelos Lgicos de Dados

Modelos lgicos baseados em objetos

descrição dos dados nos níveis conceitual e de visões de usuários.

Exemplos:

entidade- relacionamento, orientado a objetos.

No modelo orientado a objetos, código executável é parte integrante do modelo de dados.

Modelos lógicos baseados em registros

- descrição dos dados nos níveis conceitual e de visões de usuários;
- o banco de dados é estruturado em registros de formatos fixos, de diversos tipos;
- cada tipo de registro tem sua coleção de atributos;
- há linguagens para expressar consultas e atualizações no banco de dados.

Exemplos:

- relacional,
- rede,
- hierárquico.

O Modelo Relacional

O modelo relacional representa todos os dados do banco de dados em tabelas simples, mas as informações em mais de um arquivo podem ser extraídas e combinadas com facilidade. A vantagem do modelo relacional, é a de que um elemento de dado de um arquivo ou tabela pode ser relacionado a qualquer fragmento em outro arquivo desde que ambas compartilhem um elemento de dado comum.

No modelo relacional, dados e relacionamentos entre dados são representados por tabelas, cada uma com suas colunas específicas.

Exemplo das Informações em um Banco de Dados

nome rua cidade conta saldo

José Figueiras Campinas 900 55

João Laranjeiras Campinas 556 1.000

João Laranjeiras Campinas 647 5.366

Antônio Ipê São Paulo 647 5.366

Antônio Ipê São Paulo 801 10.533

Os dados são representados por coleções de registros e os relacionamentos por elos.

José Figueiras Campinas 900 55

João Laranjeiras Campinas

556 1.000

Antônio Ipê São Paulo

647 5.366

801 10.533

O Modelo de Rede

O modelo em rede é mais adequado para representar relacionamentos muitos –para –muitos entre dados . Em rede são mais flexíveis que os hierárquicos , mas os caminhos de acesso ainda precisam ser especificados antecipadamente . Existem limitações práticas para o número de ligações , ou relacionamentos, que pode ser estabelecido entre registros . Se forem excessivamente numerosos , o software funcionará eficientemente .

Os dados e relacionamentos são representados por registros e ligações, respectivamente.

Os registros são organizados como coleções arbitrárias de árvores.

José Figueiras Campinas

900 55

João Laranjeiras Campinas

556 1.000

Antônio Ipê São Paulo

647 5.366

801 10.533 647 5.366

O Modelo Hierárquico

O modelo hierárquico organiza os dados de cima para baixo , como uma árvore. Os SGBDs hierárquicos tem caminhos bem definidos e predeterminados, prestam-se mais a problemas que requerem um número limitado de respostas estruturadas que podem ser especificadas antecipadamente , são ideais para resolver problemas como o processamento diário de milhões de reservas aéreas ou de transações bancárias em caixas automáticos .

Tanto os dados quanto os relacionamentos são representados por tabelas. Possui fundamento matemático sólido. Prescinde de estruturas de índice eficientes e hardware adequado para alcançar desempenho viável em situações práticas.

Linguagem de Definição de Dados (DDL)

Permite especificar o esquema do banco de dados, através de um conjunto de definições de dados.

– A compilação dos comandos em DDL é armazenada no dicionário (ou diretório) de dados.

Linguagens de Definição e Manipulação de Dados

Manipulação de dados

- recuperação da informação armazenada,
- inserção de novas informações,
- exclusão de informações,
- modificação de dados armazenados.

Linguagem de Manipulação de Dados (DML)

Permite ao usuário acessar ou manipular os dados, vendo-os da forma como são definidos no nível de abstração mais alto do modelo de dados utilizado.

- Uma consulta ("query"), é um comando que requisita uma recuperação de informação.
- A parte de uma DML que envolve recuperação de informação é chamada linguagem de consulta.

Bibliografia

- 1 – KHOSHAFIAN, SETRAG. Banco de Dados Orientado a Objetos; traduzido por Tryte Informática. Rio de Janeiro: Infobook, 1994. 380 p.
- 2 – KORTH, HENRY F. e SILBERSCHATZ, ABRAHAM. Sistema de Banco de Dados; traduzido por Maurício Heihachiro Galvan Abe. 2ª ed. São Paulo: Makron Books, 1995. 754 p.
- 3 - YONG, Chu Shao. Banco de dados. 1ª ed. São Paulo: Atlas, 1988.
- 4 - SETZER, Valdemar W. Banco de Dados. 3 edição revista, São Paulo: Edgard Blucher, 1995, p. 10-20.
- 5 - MACHADO, Felipe, ABREU, Maurício. Projeto de Banco de Dados. 2 edição, São Paulo: Érica,

1996, p. 25-50, 208-210.

INTERNET:

1 - www.mundoo.com.br

2 - www.ulbra.tche.br/facin/po-bancodedados.html

3 - www.efagundes.com.br

4 - www.ufsc.br

"A mente que se abre a uma nova idéia jamais voltará ao seu tamanho original"

Einstein

Outros Autores: **Ana Cristina, Ana Paula, Carla Ordonó, Cristiane Vitorio e Elaine Muniz**