Estratégia Concursos

# Aula 00

*Banco de Dados p/ TJ-SC (Analista de Sistemas) - Com videoaulas - Pós-Edital*

Autor:
**Equipe Informática e TI, Thiago Rodrigues Cavalcanti**

10 de Março de 2020

**Banco de dados ..... 2**
**Apresentação do Professor ..... 2**
**Motivação para o curso ..... 3**
**Objetivos da aula ..... 5**
**Conceitos de Banco de dados ..... 5**
*Conceitos básicos ..... 5*
*Características da abordagem de BD ..... 11*
*Personagem do ecossistema de BD ..... 15*
*Evolução histórica dos SGBDs ..... 18*
**Modelo de dados e Arquitetura em três esquemas ..... 27**
*Instâncias x esquemas ..... 27*
*Modelo de Dados ..... 27*
*Arquitetura três esquemas ..... 29*
*Ambiente do Sistema de banco de dados ..... 34*
**Resumo ..... 38**
**Mapa Mental ..... 40**
**Questões comentadas ..... 41**
*Questões comentadas ..... 41*
*Exercícios ..... 58*
*Gabarito ..... 66*
**Considerações finais ..... 67**
**Referências ..... 67**





# BANCO DE DADOS

# APRESENTAÇÃO DO PROFESSOR

Olá senhoras e senhores,

Sejam bem-vindos a mais um curso de Tecnologia da Informação! Hoje damos continuidade ao mais completo curso no que se refere Banco de Dados para concursos. Gosto sempre de dizer que é um prazer imenso fazer parte desta equipe de professores do Estratégia Concursos e ter a oportunidade de apresentar um pouco do meu conhecimento e experiência em concursos públicos!

Antes de começar de fato o conteúdo teórico desta aula, vou apresentar de forma rápida. Meu nome é Thiago, sou casado, pernambucano, tenho dois filhos, Vinícius (8 anos) e Lucas (*in memoriam*). Torço pelo Sport Clube do Recife. Sou cristão. Frequento a IPN – Igreja Presbiteriana Nacional. Me formei em Ciência da Computação pela UFPE. Tenho mestrado em engenharia de software na mesma instituição. Atualmente faço doutorado em economia na UnB.

Frequento academia para manter a forma, mas meu hobby mesmo é pedalar! Decidi vender o carro e viver num desafio intermodal de transporte. Ia para o trabalho de *bike* sempre que possível! Ultimamente tenho usado mais Uber do que a magrela, mais isso é um detalhe! A pergunta é: onde eu trabalho? No Banco Central do Brasil!

Fruto de uma trajetória de dois anos de estudos diários. Aposentei as canetas em 2010. Hoje estou de licença do Banco Central para fazer doutorado que começou em março de 2017. Antes de me licenciar eu trabalhava com análise e modelagem de dados.

Minha mais recente experiência com gestão de dados é parte de uma estratégia profissional de alinhar meu trabalho diário como servidor público com minha carreira paralela de professor e consultor de Banco de Dados (BD) e *Business Intelligence* (BI). A ideia é conseguir me especializar cada vez mais no tema desta nova carreira dentro da TI, que o mercado está denominando de **cientista dos dados (*Data scientist*)**.

Entrei neste universo de professor de concurso há alguns anos. Desde 2012, tenho me dedicado especificamente ao conteúdo de BD e BI. Minhas experiências em cursos presencias aqui em Brasília e em diversas partes do Brasil, bem como na gravação sistemática de aulas on-line me ajudaram a desenvolver um conteúdo exclusivo para os alunos do Estratégia Concursos.

A ideia é desenvolver um material completo, recheado de questões e com diversas dicas para ajudar você no seu objetivo: **ser aprovado e nomeado**!




Para finalizar, não deixe de seguir minha página no Facebook® (profthiagocavalcanti), onde eu publico, sistematicamente, questões comentadas e dicas semanais. Tenho também uma conta no Instagram,

lá eu posto motivações e dicas rápidas a respeito do conteúdo de banco de dados e **análise de informações**. Agora que você já me conhece! Vamos seguir em frente com o nosso curso!

Para facilitar usa vida você pode usar os QR codes ao lado para acessar meu perfil nas redes sociais. Se precisar falar comigo por e-mail, mande mensagem para:

rcthiago@gmail.com

## MOTIVAÇÃO PARA O CURSO

Preparar esse curso é um desafio! Consolidar de forma amigável o conhecimento de banco de dados, análise de informações ou business Intelligence para concursos não é uma tarefa fácil! Calibrar o nível do teórico associado a uma didática eficiente tem sido minha meta nos últimos anos. Separamos o conteúdo de forma a segmentar e impulsionar seu aprendizado. Para que você entre na primeira aula com um pouco mais de segurança, vou aproveitar para fazer uma rápida apresentação sobre o assunto.

Você já ouviu falar sobre **Data Science ou ciência dos dados**? É um conceito relativamente recente que agrupa diversas atividades executadas sobre um conjunto de dados. Em especial sobre grandes conjuntos de dados. Para analisar os dados eles precisam estar **armazenados e organizados** de maneira **convenientes** para os cientistas dos dados. Essa base de dados facilita o trabalho e o entendimento do conteúdo armazenado.

Cientistas de dados são uma nova geração de especialistas em análise que têm habilidades técnicas para resolver problemas complexos e a curiosidade de explorar quais são os problemas que precisam ser resolvidos. A solução desses problemas passa por analisar os dados presentes em um banco de dados. Neste curso veremos o passo-a-passo para construção de um banco de dados.

Nossa primeira aula deve inserir você no universo dos bancos de dados. Um banco pode ser visto como uma estrutura que armazena algo, por exemplo, um banco de leite guarda leite guarda leite materno para que possa ser reutilizado de forma adequada em momentos posteriores. Um banco de dados guarda dados. Esses devem ser controlados de forma adequada. É nesse momento que surge um sistema para “cuidar” do acesso consistente aos dados.

Os sistemas de gerenciamento de banco de dados (SGBDs) contribuem para a disponibilidade de um conjunto de informações para diferentes usuários simultaneamente. É preciso decidir quais dados armazenar, estruturar e manter na base de dados. Para controlar esse sistema e todo o desenvolvimento do projeto e da infraestrutura associada ao sistema de banco de dados várias tarefas têm que ser feitas.

Veremos que existem profissionais dedicados a tarefas específicas. Veremos ainda que a construção de um banco de dados, em especial um banco de dados relacional, passa por algumas etapas bem definidas. Essas etapas criam modelos de dados ou esquemas que permitem um melhor entendimento da estrutura de dados da organização.

Todos esses conceitos serão vistos em detalhes nas próximas páginas. Ao final, teremos nossa tradicional lista de exercícios. Espero conseguir contribuir para a sua aprovação. Vamos em frente?!

Teremos muito trabalho! Por isso, montamos um **curso teórico em PDF,** baseado nas mais diversas bancas, apresentando o conteúdo observando as variadas formas de cobrança do mesmo pelas bancas examinadoras. Nas aulas deste curso focamos na banca FCC, podemos justificar essa escolha por ser a banca com maior número de questões recentes. Incluímos, sempre que for possível (viável) uma aula de exercício de banco de dados da sua banca ao final do curso.

Teremos ainda videoaulas que apresentam o conteúdo teórico de forma detalhada para algumas partes da matéria. Existe uma força tarefa para gravação de todo o assunto, mas não temos como garantir o término deste trabalho até a data de publicação das aulas. Mas não se preocupe, nosso objetivo é garantir que você tenha capacidade e conhecimento para ser aprovado. Logo, todo conteúdo necessário para a prova estará presente nos PDFs.

**Agora vamos voltar para a nossa aula. Vamos juntos?**

Observação importante: este curso é protegido por direitos autorais (copyright), nos termos da Lei 9.610/98, que altera, atualiza e consolida a legislação sobre direitos autorais e dá outras providências.
Grupos de rateio e pirataria são clandestinos, violam a lei e prejudicam os professores que elaboram os cursos. Valorize o trabalho de nossa equipe adquirindo os cursos honestamente através do site Estratégia Concursos ;-)

Observação importante II: todo o conteúdo deste curso encontra-se completo em nossos textos escritos. As videoaulas, caso existam, visam reforçar o aprendizado, especialmente para aqueles que possuem maior facilidade de aprendizado com vídeos e/ou querem ter mais uma opção para o aprendizado.

# OBJETIVOS DA AULA

Nossa aula começa com os objetivos. Ao final dela, ao ler o resumo, você deve ter certeza de que fixou os principais conceitos associados aos seguintes objetivos:

- Entender o que é um banco de dados e por que sistemas de banco de dados (SBD) e os sistemas de gerenciamento de banco de dados (SGBDs) são uteis.
- Entender a importância da abstração na construção de um banco de dados e como essa abstração é desenvolvida por meio de modelos de dados em diferentes níveis.
- Conhecer uma arquitetura genérica para sistemas de banco de dados denominada ANSI/SPARC.
- Descrever o ambiente de banco de dados em função dos diversos componentes.

Falaremos ainda sobre os personagens que se envolvem no funcionamento diário de um banco de dados e traçaremos uma rápida história dos Sistemas de Gerenciamento de Banco de dados.

# CONCEITOS DE BANCO DE DADOS

## CONCEITOS BÁSICOS




Antes de adentrar nos conceitos de banco de dados gostaria de voltar um pouco no tempo. Preciso que você conheça os sistemas de arquivo. Você deve ter acesso a um sistema de arquivo do computador ou dispositivo que você está usando para ter acesso a esse conteúdo. Agora imagine a seguinte situação: você trabalha em uma empresa e a lista de clientes está armazenada em um arquivo do Excel.

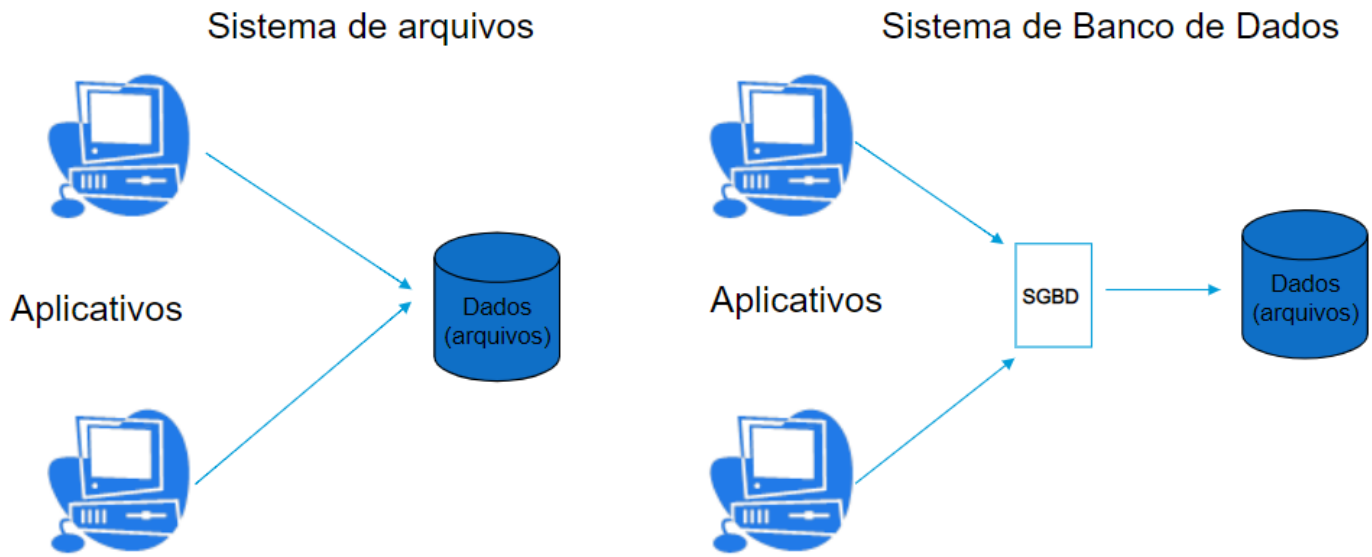
Você e todos os demais funcionários do setor de vendas (isso mesmo, na nossa história você trabalha no setor de vendas) fazem acesso ao mesmo arquivo para incluir e atualizar dados dos clientes. Quando o arquivo começa a ficar sem controle vocês resolvem criar cópias do mesmo para que cada vendedor possa ter seu próprio cadastro. A solução do problema de acesso simultâneo nos leva a outro problema: **redundância de dados**. Várias cópias sem controle podem ainda ter inconsistência entre os dados armazenados.

Para resolver o problema da **redundância não controlada**, você e seus colegas de trabalho ouvem falar de **uma solução robusta** para o problema de armazenamento de dados. Essa solução passa por manter apenas uma cópia dos dados em um banco de dados e usar um sistema gerenciador para **controlar o acesso concorrente**.

CURIOSIDADE

Há dois tipos de redundância de dados:
**Redundância controlada de dados:** Acontece quando o software tem conhecimento da múltipla representação da informação e garante a sincronização entre as diversas representações.
**Redundância não controlada:** Acontece quando a responsabilidade pela manutenção da sincronia entre as diversas representações de uma informação está com o usuário e não com o software.

No universo tecnológico o que aconteceu em um passado recente foi mais ou menos a mesma coisa. Aplicações empresariais faziam acesso a arquivos sem um elemento central para controlar o processo. Eis que surge o Sistema de Gerenciamento de Banco de Dados (SGBD) para atuar como um guardião do banco de dados, que substituiu a abordagem de arquivos. Vejamos uma figura para esclarecer esses contextos:




Perceba que do lado esquerdo não temos o SGBD, já do lado direito existe o elemento. Para entender como e por que esse camarada apareceu aí vamos compreender os conceitos primordiais para o entendimento do assunto. Em qualquer ciência, o entendimento completo do seu conteúdo deve se basear nos conceitos fundamentais. Nossa primeira aula começa focada nesses conceitos.

Quando iniciamos nosso estudo, precisamos ter a capacidade de responder a seguinte pergunta:

O que é **banco de dados**?

Você já consegue definir esse termo e suas principais características? Se você ainda não tem o entendimento perfeito do que seria um banco de dados, nosso objetivo agora é construir o conceito. Uma forma tradicional de definir banco de dados é começar entendendo o significado das palavras que compõe o termo: **Banco** e **dados**.

**Banco** tem diversas definições possíveis no dicionário da língua portuguesa. Dentre elas a que melhor se encaixa no nosso contexto é **um conjunto organizado e categorizado de objetos**, por exemplo, podemos ter um banco de fotografias ou um banco de leite.

**Dados** são **fatos conhecidos** que podem ser registrados e possuem **um significado implícito**. Esse conceito, porém, é um pouco amplo e abstrato para nosso intuito. Quando reduzimos o escopo à tecnologia da informação, temos um conceito mais adequado para dado. Ele é a representação física de um evento no tempo e espaço que não agrega fundamento ou significado para quem o sente ou recebe. É, basicamente, **um registro**!

Imagine que eu fale para você por "32260436, cinco, aprovado, Thiago". Você vai pensar, o professor está ficando louco! Mas em um banco de dados, esses registros são armazenados e chamados de dados. Para representarem uma informação eles precisam de um contexto associado. Se pensarmos em uma agenda telefônica o número "32260436" pode ser o telefone de Thiago. Aprovado pode ser a sua situação no seu próximo concurso.

Agora que temos o entendimento dos termos vamos partir para a definição do banco de dados.



De forma simples e direta: um **Banco de dados** é uma coleção de **dados** relacionados. Vejam que essa definição não estabelece a necessidade dos dados serem armazenados em formato digital. Alguns livros trazem o exemplo de uma agenda telefônica de papel como um exemplo bastante didático do conceito de banco de dados.

*O banco de dados, por si só, pode ser considerado como o equivalente eletrônico de um armário de arquivamento; ou seja, ele é um repositório ou recipiente para uma* ***coleção de arquivos de dados computadorizados****. – C J Date*

Esta definição, porém, é considerada muito simplista para alguns autores por não contextualizar o termo. O *Navathe*, por exemplo, cita três **propriedades implícitas** que contribuem para o entendimento do termo banco de dados. Primeiramente, ele representa **algum aspecto do mundo real**, às vezes chamado de **minimundo** ou de **universo de discurso** (*UoD – Universe of Discourse*). As mudanças no minimundo devem ser refletidas no banco de dados.

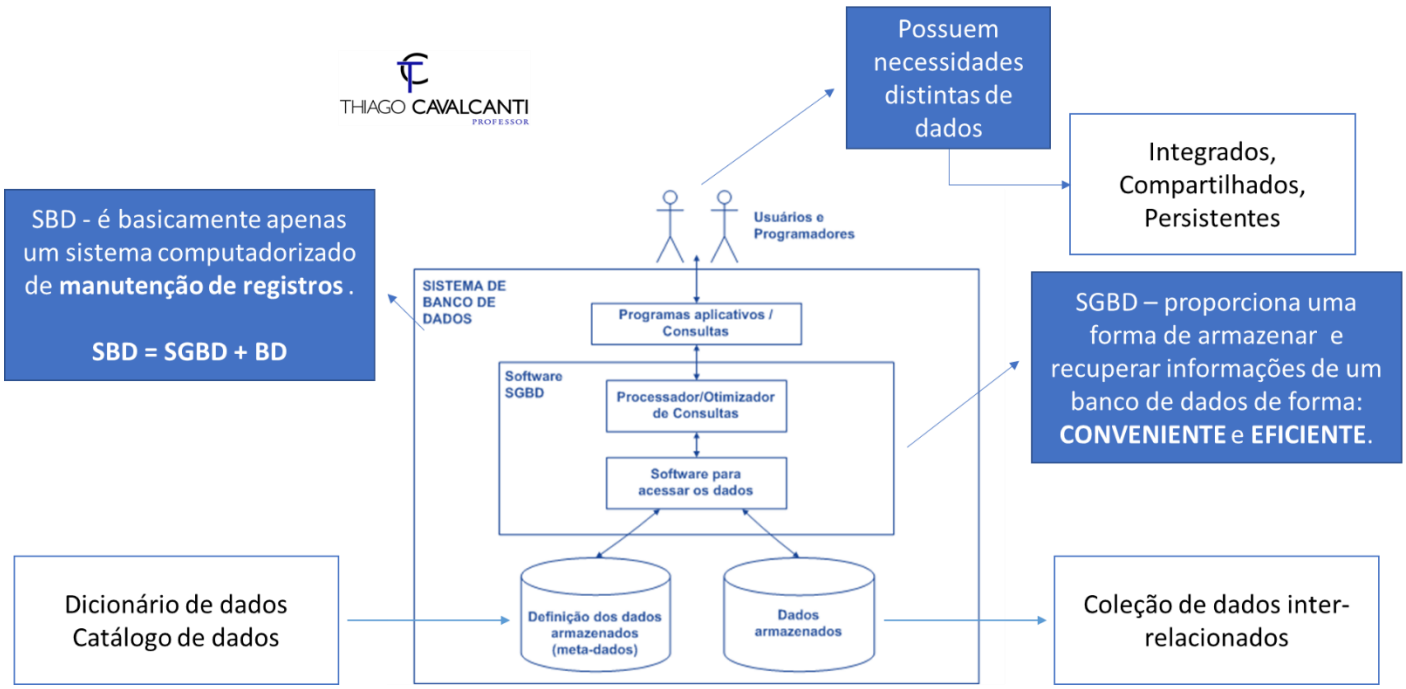
A **segunda** característica implícita diz que a **coleção de dados é logicamente coerente** com algum significado inerente. Uma variedade aleatória de dados **não** pode ser chamada de banco de dados. Um banco de dados pode armazenar as informações de uma empresa, uma faculdade ou um órgão do setor público. Veja que essas informações estão dentro de um contexto, sendo, logicamente coerente.

A **terceira** propriedade afirma que um banco de dados é construído e populado com dados para uma **finalidade específica**. Ele possui um grupo de **usuários** bem definido e algumas aplicações, previamente concebidas, sobre as quais esses usuários interessados fazem acesso aos dados. Não adianta você criar um repositório sem propósito! Ele tem que atender alguma necessidade de acesso ao conjunto dos dados.

Vamos voltar ao nosso exemplo físico da agenda telefônica, você consegue visualizar as propriedades definidas pelo Navathe neste banco de dados. Qual o aspecto do mundo real que ela representa? Os dados estão logicamente relacionados? Possuem uma finalidade específica? Acredito que sim! Tente responder a essas perguntas mentalmente antes de seguir em frente.

As três propriedades definidas pelo Navathe começam a estruturar os conceitos na sua cabeça. Outra definição de **banco de dados** que resume o que apresentamos até agora é: "um conjunto de dados **estruturados** que são confiáveis, coerentes e compartilhados por usuários que têm necessidades de informações diferentes".

O primeiro conceito básico você já dever ter entendido. Você não deve ter dúvidas sobre o que seria um banco de dados. Vamos agora entender a diferença entre **banco de dados**, **sistemas de gerenciamento de banco de dados (SGBD)** e **sistemas de banco de dados (SBD)**. São três conceitos diferentes para os autores dos livros teóricos sobre o assunto. Para entender essas diferenças peço que você observe a figura a seguir:




Seguindo o fluxo de acesso aos dados, podemos observar que os usuários e programadores se comunicam com o sistema de banco de dados. Este, por sua vez, faz acesso ao sistema de gerenciamento do banco de dados. O SGDB usa as informações presentes nos bancos de dados, representados pelos cilindros da figura acima, para ter acesso aos dados armazenados.

Um **Sistema de Gerenciamento de Banco de Dados (SGBD)** é um **conjunto de programas** que permitem armazenar, modificar e extrair informações de um banco de dados. Seu **principal objetivo** é proporcionar um ambiente tanto conveniente quanto eficiente para a **recuperação e armazenamento** das informações do banco de dados.

Contudo, os SGBDs não se restringem apenas a manipulação dos dados. Eles fornecem uma variedade de programas com diferentes funcionalidades.




A figura acima apresenta algumas funcionalidades dos SGBDs. Cada espaço está preenchido com alguma funcionalidade que faz parte do escopo de um software de SGBD. Não se preocupe se você não tiver ideia do que está presente em cada espaço, ao longo desta aula ou do nosso curso, quando você começar a entender melhor o assunto, você perceberá a presença deles dentro do contexto de um SGBD. Vamos em frente!

Outro aspecto interessante sobre banco de dados são as descrições ou definições dos objetos, pense em uma tabela do modelo relacional. Para termos acesso a uma tabela precisamos conhecer sua estrutura, começando pelo seu nome e das suas colunas. Essa tarefa envolve especificar os **tipos, estruturas** e **restrições** dos dados a serem armazenados.

A definição ou informação descritiva do banco de dados também é armazenada pelo SGBD numa estrutura conhecida como **catálogo** ou **dicionário de dados**, que armazena os chamados de **metadados** dos objetos**.** Os metadados carregam consigo um significado. Uma coluna de uma tabela pode ser definida por um tipo de dados, inteiro, uma restrição, *not null*.

É possível ainda fazer o **compartilhamento dos dados** entre diversos usuários e programas, possibilitando o acesso ao banco de dados de forma simultânea. Outras funções importantes também são providas como **proteção** do sistema **contra defeitos** de hardware e software, feitos por meio de redundância ou replicação, e **proteção** de segurança **contra acesso** não autorizados ou maliciosos.

Outros aspectos interessantes estão relacionados com o controle de transações, recuperação após falha, otimização de consultas ou do próprio SGBD, auditoria por meio de logs de sistema, enfim, são várias as **funcionalidades providas** pelos softwares presentes em um **SGBD**. Vamos agora definir o próximo conceito: sistema de banco de dados!

O **sistema de banco de dados (SBD)** é considerado a união entre o banco de dados e o sistema de gerenciamento de banco de dados. Em outras palavras, consiste em uma coleção de dados inter-

relacionados e de um conjunto de programas para acessá-los. Partindo da figura que apresentamos anteriormente conseguimos construir a seguinte fórmula:

**SBD = BD + SGBD + (Programa de aplicação/consulta)**

Antes de continuarmos construindo nosso arcabouço teórico sobre o assunto vamos resolver duas questões de provas passadas.




**1. Ano: 2016 Órgão: TCE-SC Prova: Auditor Fiscal de Controle Externo - Informática**

Com relação aos bancos de dados relacionais, julgue o próximo item.

O catálogo de um sistema de gerenciamento de banco de dados relacional armazena a descrição da estrutura do banco de dados e contém informações a respeito de cada arquivo, do tipo e formato de armazenamento de cada item de dado e das restrições relativas aos dados.

**Comentário:** Perceba que a definição acima está de acordo com o termo dicionários de dados, catálogo de dados ou metadados presentes em um sistema de banco de dados. Lembre-se que essa separação entre a descrição dos dados e os dados propriamente dito é uma das características relevantes que foram apresentadas na evolução de sistemas de arquivos para a **abordagem de banco de dados**. Sendo assim, podemos afirmar que a questão está correta!

**Gabarito: C.**

---

**2. Ano: 2010 Órgão: Banco da Amazônia Prova: Técnico Científico - Tecnologia da Informação**

O dicionário de dados é uma das principais ferramentas para a administração dos dados corporativos. Por meio da engenharia reversa, pode-se armazenar os modelos de dados, as estruturas de dados, seus relacionamentos e toda a documentação necessária para garantir facilidade na localização e manipulação dos dados. Acerca dos papéis do administrador de dados (AD) e dos dicionários de dados, julgue os itens a seguir.

[1] O dicionário de dados é considerado um subconjunto das funções de um catálogo de sistema.

[2] O catálogo do sistema é um repositório com função de armazenar as definições dos esquemas dos bancos de dados.

**Comentário:** É importante lembrar que existe uma hierarquia entre os objetos ou elementos em um dicionário de dados. Um dicionário de dados possui a descrição dos esquemas ou catálogo de sistemas. Cada catálogo deve conter a descrição dos objetos que fazem parte do contexto de um sistema, como tabelas, visões e domínios. Dentro das definições das tabelas temos as descrições do atributos e restrições de integridades dos dados.

Assim, ao analisar as alternativas acima, podemos inferir que a alternativa [1] encontra-se incorreta, já a afirmação [2] está certa!

**Gabarito: E C.**

## CARACTERÍSTICAS DA ABORDAGEM DE BD

Segundo Navathe, são quatro, as principais características da abordagem de banco de dados que a fazem sobressair em relação às abordagens de processamento de arquivo.

> **Natureza de autodescrição** de um sistema de banco de dados
> Isolamento entre programas e dados, **abstração de dados**
> Suporte a **múltiplas visões** de dados
> **Compartilhamento** de dados e processamento de transação multiusuário.

Esses esforços visam **reduzir a redundância** o que implica em reduzir o desperdício no espaço de armazenamento e os esforços para manter os dados comuns ou duplicados atualizados. Tudo realizado por meio de um único repositório!

Vejam que a lista acima pode ser caracterizada como uma enumeração e, como eu sempre digo, listas fazem parte do rol de questões de prova de concurso. Seja qual for a matéria, sempre gaste um pouco do seu tempo lendo, mais de uma vez, cada uma das listas pertencentes aos assuntos. A verdade é: não importa o grau de relevância dentro do assunto, um examinador preguiçoso sempre está propício a utilizar deste artifício ao elaborar uma questão.

A primeira característica listada pelo Navathe é conhecida por nós como **catálogo do SGBD**, **dicionário de dados ou metadados**. Esta propriedade permite ao SGBD gravar as definições das suas estruturas e restrições. E, quais são as descrições que podem ser gravadas? Descrição de tabelas, tamanho do campo, tipo dos dados, propriedade de ser nulo ou não, valores default, restrições de integridade, entre outros. Para facilitar sua visualização pense numa definição de uma tabela em SQL[1]. Veja o exemplo a seguir e observe algumas dessas descrições.

```
CREATE TABLE PRL_EMPLOYEE(
  ID_EMPLOYEE NUMBER,
  FK_ID_MANAGER NUMBER,
  EMPLOYEE_FIRST_NAME VARCHAR2(100) NOT NULL,
  EMPLOYEE_LAST_NAME VARCHAR2(100) NOT NULL,
  EMPLOYEE_EMAIL VARCHAR2(100) NOT NULL,
  EMPLOYEE_B_DATE DATE NOT NULL,
  EMPLOYEE_START_DATE DATE DEFAULT SYSDATE,
  EMPLOYEE_END_DATE DATE DEFAULT NULL,
  EMPLOYEE_LOCKED NUMBER(1) DEFAULT 1
)
```

A próxima característica é uma decorrência da anterior. A partir do momento em que temos um dicionário de dados, é possível excluir da estrutura dos programas a definição dos dados presentes

[1] SQL é uma linguagem declarativa usada para enviar comandos aos sistemas de gerenciamento de banco de dados. Esses comandos permitem a criação de tabelas e a manipulação dos dados armazenados. Na criação da tabela podemos observar a definição dos tipos de dados (VARCHAR2, DATE, NUMBER), bem como das restrições de integridade (NOT NULL, DEFAULT).

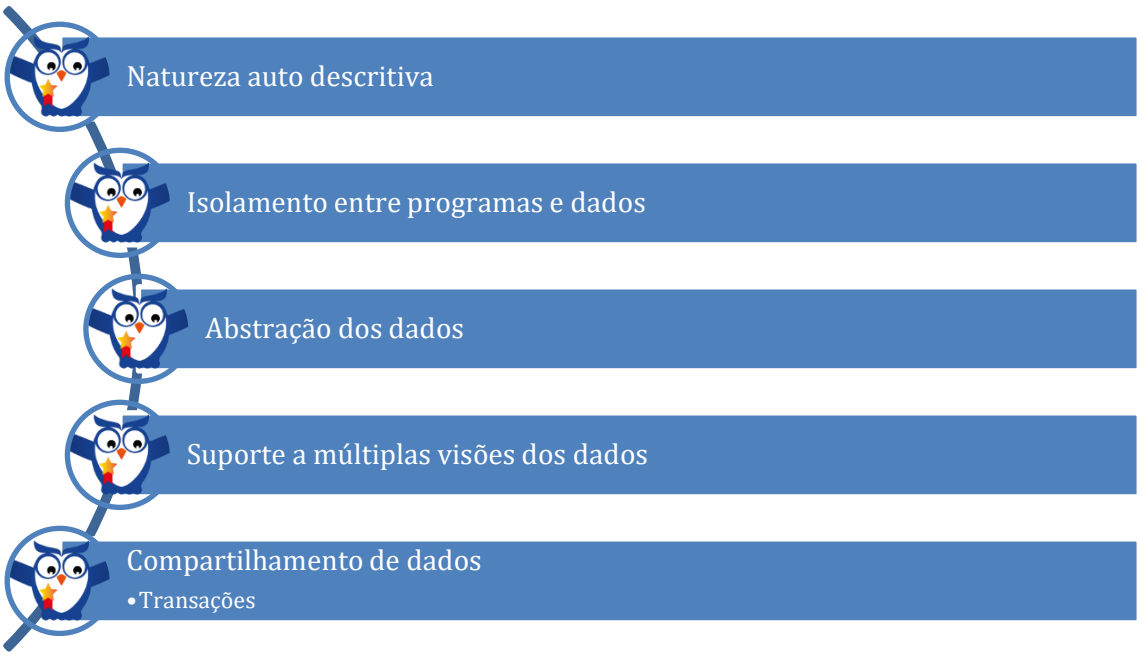
nos mesmos. Agora isolados, dados e aplicações, criam um conceito chamado **independência de dados do programa**. Este só é possível por conta da **abstração de dados**. A abstração de dados permite a criação de diferentes níveis de modelos. Cada modelo revela a estrutura dos dados de uma forma específica.

O suporte a **múltiplas visões** parte do princípio que diferentes usuários têm diferentes necessidades sobre os dados. Se pensarmos em SQL, uma VIEW representa um subconjunto de informações referentes a uma ou mais tabelas (ou até a nenhuma tabela). Do ponto de vista mais abstrato, uma visão é a parte do banco de dados ao qual um usuário ou grupo de usuários tem acesso. Porém, existe a possibilidade dessa visão conter um **dado virtual** que é derivado das informações armazenadas. Imagine, por exemplo, a idade calculada a partir da data de nascimento.

Quando falamos de suporte a **múltiplos usuários** queremos, basicamente, permitir que diferentes usuários acessem o banco de dados ao mesmo tempo. Para garantir que isso ocorra é preciso que o SGBD forneça um mecanismo de **controle de concorrência**. As transações efetuadas devem levar o sistema a um estado válido (C), não ter conhecimento umas das outras (I), serem executadas sempre por completo (ou não serem executadas) (A) e, uma vez gravadas na base, devem persistir ao longo do tempo (D).

Você dever estar se perguntando, o que são essas letras ao lado das características das transações citadas anteriormente? Elas se referem às características de uma transação: **atomicidade**, **consistência**, **isolamento** e **durabilidade**. Elas formam uma sigla conhecida como ACID e são bastante conhecidas dentro do modelo de dados relacional/transacional.



Acabamos de tratar das características que o Navathe utiliza para diferenciar sistemas de arquivo dos sistemas de banco de dados. Vamos agora listar as características descritas pelo Date e pelo Silberschatz. Date chama de benefícios da abordagem de banco de dados. Quais sejam:

1. O dado pode ser compartilhado
2. A redundância pode ser reduzida
3. Inconsistências podem ser evitadas
4. Pode-se utilizar o suporte a transações
5. A integridade pode ser mantida
6. A segurança pode ser aperfeiçoada
7. Requisitos conflitantes podem ser balanceados
8. Padrões podem ser utilizados

Já Abraham Silberschatz trata das desvantagens de se utilizar um **sistema de arquivo**:

1. Redundância e inconsistência dos dados
2. Dificuldade de acesso a dados
3. Isolamento dos dados
4. Problemas de integridade
5. Problemas de atomicidade
6. Anomalias de acesso concorrente
7. Problemas de segurança

Lembrem-se, não precisamos decorar todas essas listas, apenas tomar conhecimento da sua existência, pois fazem parte do contexto. Elas procuram sempre expor as características que diferenciam os sistemas de arquivos dos sistemas de banco de dados.

Vejamos mais algumas questões que tratam do assunto. Começando por uma questão do TCE-PB.

CAIU na prova!

**3. Ano: 2018 Órgão: TCE-PB Cargo: Auditor de Constas Públicas Questão: 97**

A respeito de SGBDs, assinale a opção correta.

A Um SGBD, por definição, não é flexível, dada a dificuldade de mudar a estrutura dos dados quando os requisitos mudam.

B Um SGBD é um software que não prevê as funções de definição, recuperação e alteração de dados, sendo essa tarefa a função básica de um sistema de banco de dados.

C A consistência de dados é o princípio que determina a manutenção de determinado dado em vários arquivos diferentes.

D Conforme o princípio da atomicidade, caso ocorra erro em determinada transação, todo o conjunto a ela relacionado será desfeito até o retorno ao estado inicial, como se a transação nunca tivesse sido executada.

E O controle de concorrência é o princípio que garante e permite a manipulação, no mesmo momento, de um mesmo dado por mais de uma pessoa ou um sistema.

**Comentário:** Vamos analisar as alternativas acima. Elas são relevantes para consolidar nosso conhecimento sobre o assunto. Começando pela alternativa A, o erro aparece quando o examinador afirmar que os SGBDs não são flexíveis. Lembre-se que um SGBD veio solucionar um problema de replicação dos dados em diversos arquivos distintos. Agora, com uma fonte única dos dados, as alterações dos mesmos são centralizadas.

Já a alternativa B nos remonta as funcionalidades do SGBD. O que podemos fazer com tal sistema? É possível, primariamente, armazenar e manipular dados, para tal, é necessário descrever as estruturas das tabelas. Todos os SGBDs possuem estrutura para construção do banco de dados, usando uma linguagem como SQL. Sendo assim, a alternativa B também está incorreta.

As próximas duas alternativas tratam das propriedades das transações, mais especificamente da consistência, que tem por objetivo levar o banco de dados de um estado válido para outro estado consistente, e da atomicidade, que parte do princípio que uma transação é composta por vários comandos de modificação da base de dados e que estes comandos devem ser executados em conjunto completamente ou não serem executados. Desta forma, para garantir a atomicidade, caso uma transação falhe todas as operações já efetivadas precisam ser desfeitas. Desta forma, a alternativa C está incorreta e a alternativa D é a nossa resposta.

Por fim, o controle de concorrência é um mecanismo que permite que apenas um usuário consiga modificar um dado do sistema em um determinado momento. É importante entender que é possível várias pessoas terem acesso aos dados para leitura. Como se todos estivessem lendo um livro. Contudo, na hora de escrever, só existe uma caneta e, se alguma pessoa estiver de posse da caneta, apenas ela poderá fazer alterações na base.

**Gabarito: D.**

---

**4. ANO: 2014 ÓRGÃO: TJ-AP PROVA: ANALISTA JUDICIÁRIO - BANCO DE DADOS - DBA**

A redundância controlada de dados em um sistema de banco de dados ocorre quando

A um programa está ciente da múltipla representação de uma dada informação e garante o sincronismo entre as diversas representações.

B a responsabilidade pela manutenção do sincronismo entre as múltiplas representações de uma dada informação é compartilhada entre o programa e o usuário.

C os dados mais importantes são duplicados a cada backup do sistema, visando aumentar a garantia da recuperação da informação em caso de problemas.

D a responsabilidade pela manutenção do sincronismo entre as múltiplas representações de uma dada informação é do usuário.

E um programa está ciente da múltipla representação de uma dada informação, mas não garante o sincronismo entre as diversas representações.

**Comentário:** Nesta questão vamos entender o que está sendo cobrado antes de avaliarmos as alternativas. Em primeiro lugar, precisamos lembrar que o SGBD vai controlar a redundância dentro do banco de dados. Esse controle é feito quando diferentes usuários compartilham a mesma informação.

Suponha uma tabela de endereços de clientes em um banco comercial. Todos os setores do banco podem fazer uso desta informação. A área de investimento pode mandar o extrato das suas aplicações, o setor de cartão de crédito pode enviar sua fatura e o relacionamento com o cliente pode te enviar um cartão de feliz aniversário. A importância deste ponto único de contato é verificada quando a cliente muda de endereço, a atualização da tabela vai ser efetiva para os diferentes usuários da informação.

Agora vamos analisar as alternativas, ao ler cada uma delas, podemos observar que no SGBD um programa deve estar ciente da múltipla representação de uma dada informação e garantir o sincronismo entre as diversas representações. Essa descrição é a presentes na alternativa A que é a nossa resposta.

**Gabarito: A.**

## PERSONAGEM DO ECOSSISTEMA DE BD

Quando tratamos de grandes organizações, as atividades relacionadas a banco de dados devem ser compartilhadas entre diferentes pessoas. Trataremos agora dos dois principais papéis dentro desse processo: o administrador de banco de dados (DBA) e o administrador de dados (AD).

Só para termos uma ideia, algumas empresas do setor bancário chegam a ter algumas dezenas de ADs dentro da organização. O Bradesco tem por volta de 40 ADs. Vamos então começar falando um pouco sobre esse perfil de trabalhador especializado em banco de dados.

O AD é a pessoa que toma as decisões estratégicas e de normas com relação aos dados da empresa. Os **administradores de dados** também podem ser conhecidos por projetista de dados. Suas tarefas são realizadas principalmente antes do banco de dados ser realmente implementado e/ou populado.

Eles são responsáveis por identificar os dados a serem armazenados e escolher estruturas apropriadas para representar esses dados. Para isso precisam se comunicar com todos os potenciais usuários a fim de entender suas necessidades e criar um projeto que as atenda suas necessidades. Eles definem então visões para cada grupo de usuários. Podemos listar ainda como atribuições do AD:

- Padronizar os nomes dos objetos criados no BD
- Gerenciar e auxiliar na definição das regras de integridade
- Controlar a existência de informações redundantes
- Trabalhar de forma corporativa nos modelos de dados da organização

Falaremos agora do **DBA – *Database Administrator***, ou, da pessoa que fornece o **suporte técnico** necessário para implementar as decisões. Assim, o DBA é responsável pelo controle geral do sistema em um nível técnico. Tem como **recurso primário** o banco de dados e como **recursos secundários** o SGBD e os softwares relacionados.

O DBA é o responsável por autorizar o acesso ao banco de dados, coordenar e monitorar seu uso, adquirir recursos de software e hardware conforme a necessidade e por resolver problemas tais como falhas de segurança ou demora no tempo de resposta do sistema. Segundo o Date uma lista de atividades associadas ao DBA contém as seguintes tarefas:

- Definir o esquema conceitual (às vezes conhecido como lógico)
- Definir o esquema interno
- Contatar com os usuários
- Definir restrições de segurança e integridade
- Monitorar o desempenho e responder a requisitos de mudanças.
- Definir normas de descarga e recarga (dumping)

Vejamos duas questões rápida sobre o assunto:




**5. Ano: 2018 Órgão: EBSERH Prova: Analista de Tecnologia da Informação**

Com relação a banco de dados, julgue o item seguinte.

Após um banco de dados ser criado, o administrador executa uma série de tarefas para dar permissão de acesso aos usuários que necessitam ler e gravar informações na base de dados. A responsabilidade de gerir os acessos ao banco de dados é do sistema gerenciador de banco de dados (SGBD).

**Comentário:** Observe que neste texto podemos verificar explicitamente o benefício do sistema de gerenciamento de banco de dados. O administrador de banco de dados vai atribuir aos diversos usuários as permissões compatíveis com suas necessidades. Contudo, quem vai gerenciar o acesso, no sentido de autorizar o usuário a acessar o banco de dados e visualizar os dados, é o próprio SGBD. Desta forma, temos uma alternativa correta.

**Gabarito: C.**

---




**6. ANO: 2010 ÓRGÃO: TCE-SP PROVA: AGENTE DA FISCALIZAÇÃO FINANCEIRA - PRODUÇÃO E BANCO DE DADOS**

Considerando que os dados constituem um dos bens mais valiosos de uma empresa, é necessário que haja um papel que tenha a responsabilidade central pelos dados,

principalmente entendendo as necessidades empresariais nos altos níveis da organização. Tal papel é mais adequadamente desempenhado pela

A administração de banco de dados.

B administração de dados.

C alta administração da organização.

D análise de sistemas.

E gerência de sistemas.

**Comentário:** Percebam que a questão trata da padronização dos dados corporativos. Entender a organização e sua relação com as informações de forma a gerenciar os dados de maneira eficiente é reponsabilidade da equipe de **administração de dados**. Hoje em dia um termo muito utilizado para os dados que permeiam vários setores das empresas ou órgãos públicos é o conceito de **dados mestres**. Eles são armazenados em um repositório central e distribuídos aos diversos usuários.

**Gabarito: B.**

---

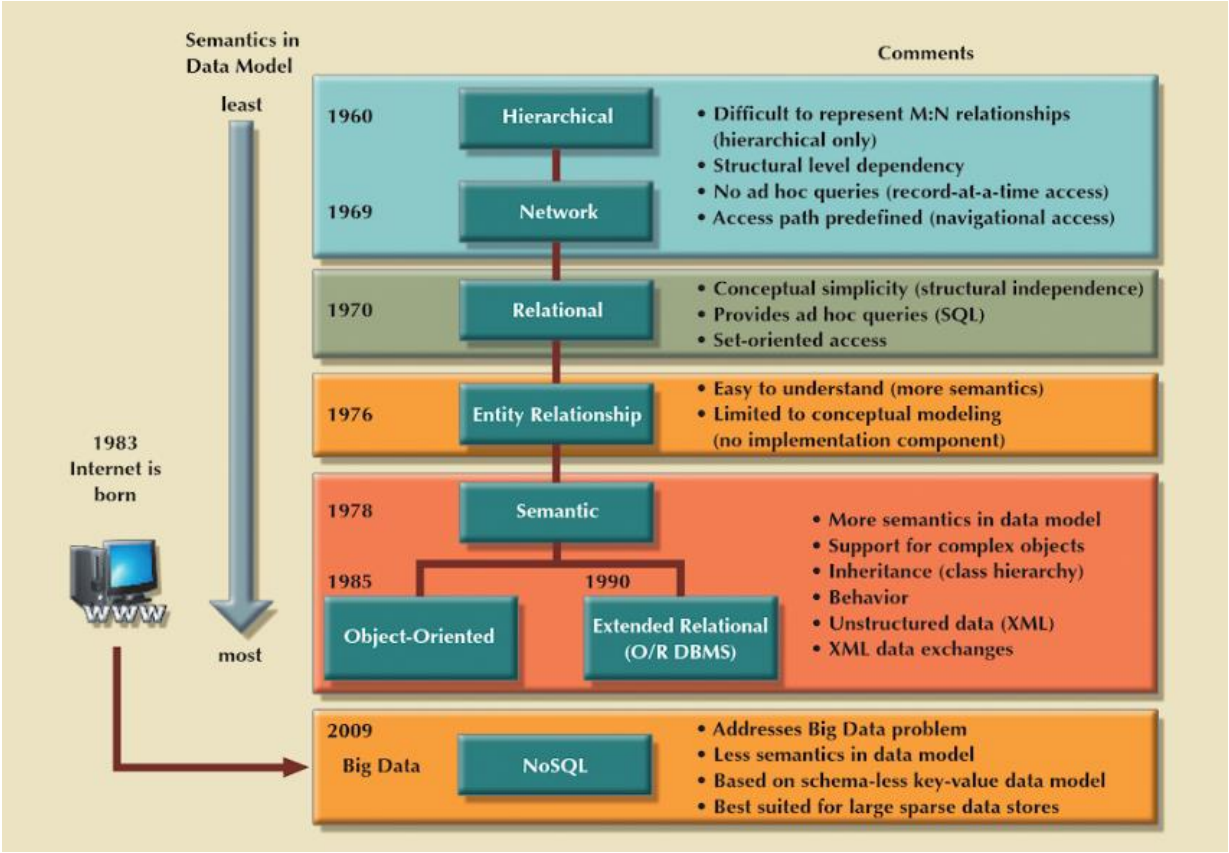
Você precisa entender que são dois os principais papeis presentes no trabalho diário de organização e manutenção de banco de dados. O **administrador de dados** que cuida do contexto estratégico dos dados, sua preocupação está em saber quais os dados serão armazenados em um banco de dados, quem são as pessoas que precisam dos mesmos e ainda, procurar padronizar os nomes dos atributos e outros objetos para que dados redundantes não sejam armazenados.



O outro indivíduo importante neste processo é o **administrador de banco de dados**. Nessa função encontramos todo o suporte técnico necessário para as atividades diárias de um sistema de banco de dados. O DBA distribui as permissões de acesso aos dados, criar os objetos que vão armazenar os dados, faz ajustes para que os mesmos funcionem de forma adequada.

## EVOLUÇÃO HISTÓRICA DOS SGBDS

É interessante conhecer a evolução dos modelos de dados até os bancos de dados NoSQL. Afinal, quando começamos a tratar as informações em sistemas como elas eram armazenadas? Veja a figura abaixo para conhecer um pouco mais sobre os modelos de armazenamento de dados.




Os primeiros sistemas de gerenciamento de banco de dados são implementados no final da década de 1960. Charles Bachmann desenvolveu o primeiro SGBD chamado *Integrated Data Store* (IDS) enquanto trabalhava na Honeywell. Esse sistema usava o **modelo de rede** onde as relações de dados são representadas como um **grafo bidirecional**.

Contudo, o primeiro SGBD que obteve sucesso comercial foi desenvolvido pela IBM chamado *Information Management System* (IMS). Ele usava o **modelo hierárquico** no qual as relações entre os dados são representadas como **uma árvore**. Por incrível que pareça, ainda está em uso hoje no sistema de reservas SABRE da IBM na American Airlines. Nesta época a Conference On Data Systems Languages (CODASYL) definiu um modelo de rede mais padronizado.

Esses dois modelos, em rede e hierárquico, apresentavam problemas sérios, entre eles:

- O acesso ao banco de dados feito através de operações com o ponteiro de baixo nível.
- Detalhes de armazenamento dependiam do tipo de dados a serem armazenados.

- Para adicionar um campo no banco era necessário reescrever o esquema subjacente de acesso/modificação, em outras palavras o modelo de dados físico.
- Ênfase nos registros a serem processados, não na estrutura global.
- O usuário tinha que conhecer a estrutura física da BD, para fim de consulta das informações.

No geral os primeiros SGBDs eram muito complexos e inflexíveis, o que tornou cada vez mais difícil o trabalho, quando era necessária a adição de novos aplicativos ou a reorganização dos dados. Para resolver esses e outros problemas **Edgar (Ted) Codd**, conhecido com o pai do **modelo relacional**, trabalhando no laboratório da IBM em San Jose propôs no artigo "A Relational Model of Data for Large Shared Data Banks" a definição do modelo relacional.

Segundo Codd, o modelo fornece um meio de descrição de dados apresentando apenas a sua estrutura natural - isto é, sem sobreposição de qualquer estrutura adicional para efeitos de representação física dos dados. Assim, ele forneceu uma base para uma linguagem de dados de alto nível que permite obter a **independência** máxima **entre dados e programas** com a **representação** de um lado e a **estrutura física** da máquina do outro.

Em outras palavras, o modelo relacional consistiu na independência de dados e, na forma de acesso aos dados definida por uma linguagem. Em vez de processar um registro de cada vez, um programador pode usar o idioma para **especificar operações** individuais que seriam **realizados** em todo o **conjunto de dados**.

Devido à natureza técnica do artigo e a relativa complicação matemática presente no texto, o significado e proposição do artigo não foram prontamente identificados. Entretanto ele levou a IBM a montar um grupo de pesquisa conhecido como **System R** (Sistema R).

O projeto do Sistema R era criar um sistema de banco de dados relacional o qual eventualmente se tornaria um produto. Os primeiros protótipos foram utilizados por muitas organizações, tais como na *Sloan School of Management* (renomada escola de negócios norte-americana). Novas versões foram testadas com empresas de aviação para rastreamento de manufaturas em estoque.

Eventualmente o Sistema R evoluiu para SQL/DS, o qual posteriormente tornou-se **o DB2**. A **linguagem** criada pelo grupo do Sistema R foi a ***Structured Query Language*** (SQL) ou linguagem de consulta estruturada. Esta linguagem tornou-se um padrão na indústria para bancos de dados relacionais e, hoje em dia, é um padrão ISO (*International Organization for Standardization*). A linguagem SQL era originalmente conhecida como SEQUEL (*Structured English QUEry Language*). Depois teve seu nome modificado para SQL por problemas de patentes.

Em meados da década de 80 tornou-se óbvio que existiam várias **áreas onde bancos de dados relacionais não eram aplicáveis**, por causa dos tipos de dados envolvidos. Estas áreas incluíam medicina, multimídia e física nuclear, todas com necessidades de flexibilidade para definir como os dados seriam representados e acessados.

Este fato levou ao início de pesquisas em **bancos de dados orientados a objetos**, nos quais os usuários poderiam definir seus próprios métodos de acesso aos dados e como estes seriam representados e acessados. Ao mesmo tempo, linguagens de programação orientadas a objetos (*Object Oriented Programming* - POO), tais como C++, começaram a surgir na indústria.

No início de 1990, temos a aparição do primeiro Sistema de Gerenciamento de Banco de Dados Orientado a Objetos (SGBDOO), através da companhia *Objectivity*. Isso permitiu que usuários criassem sistemas de banco de dados para armazenar resultados de pesquisas como o CERN (maior laboratório que trabalha com partículas em pesquisas de física nuclear - europeu) e SLAC (Centro de Aceleração Nuclear - norte-americano), para mapeamento de rede de provedores de telecomunicações e para armazenar registros médicos de pacientes em hospitais, consultórios e laboratórios.

A evolução nos levou aos SGBDs orientados a objetos, mas a praticidade nos trouxe de volta para o modelo objeto-relacional, pois a grande maioria das empresas continuou utilizando os bancos de dados relacionais. Contudo esse modelo começou a apresentar outra lista de problemas ou desafios:

1. Dados na ordem de dezenas ou centenas de TB – abordagem de cluster é cara.
2. Poder de crescimento elástico horizontal – controle de transação ACID torna inviável com a elasticidade.
3. Fácil distribuição dos dados e/ou processamento – SGBD paralelos são caros.
4. Tipos de dados variados, complexos e/ou semiestruturados – modelo de dados objeto-relacional não resolve todos os requisitos.

Tivemos então o surgimento de um novo movimento no mercado em busca de uma solução que superasse tais problemas: **o movimento NoSQL**. Este teve sua origem em junho de 2009, para nomear um encontro promovido por **Johan Oskarsson** e **Eric Evans**, que teve como objetivo discutir o surgimento crescente de **soluções *open source* de armazenamento de dados distribuídos** não relacionais.

Podemos considerar NoSQL uma nova onda de SGBDs, pois propõe algumas alternativas ao modelo relacional, porém com uma grande diferença histórica: o movimento NoSQL não tem como objetivo invalidar ou promover a total substituição do modelo relacional, e sim o fim do modelo relacional como bala de prata, como a única solução correta ou válida. Inclusive, é importante entender que NoSQL não significa “no SQL” (não ao SQL), mas sim “not only SQL” (não só SQL).

> ***Curiosidade**: Ao que tudo indica o termo NoSQL foi criado em 1998 por Carlo Strozzi para nomear seu projeto open source, que tinha como objetivo ser uma implementação mais leve de um banco de dados relacional, porém sua principal característica era não expor a interface SQL. Portanto é bem irônico usar o termo NoSQL, criado para nomear um banco de dados relacional, para classificar soluções de armazenamento de dados não relacionais.*

Juntamente com NoSQL surge o conceito de Big Data. A definição mais tradicional usa a equação dos cinco Vs. Nela, Big Data = volume + variedade + velocidade + veracidade + valor, de dados. Cada característica desta é associada aos dados armazenados. É possível justificar cada uma delas. Vejamos:

**Volume** porque além dos dados gerados pelos sistemas transacionais, temos a imensidão de dados gerados pelos objetos na Internet das Coisas, como sensores e câmeras, e os dados gerados nas mídias sociais por meio de PCs, smartphones e tablets.
**Variedade** porque estamos tratando tanto de dados textuais estruturados como não estruturados como fotos, vídeos, e-mails e tuites.
**Velocidade** porque muitas vezes precisamos responder aos eventos quase que em tempo real, ou seja, estamos falando de geração e tratamento de dados em fluxo massivos.

O ponto de vista da **veracidade** também deve ser considerado, pois não adianta muita coisa lidar com a combinação "volume + velocidade + variedade" se não houver dados confiáveis. É necessário que haja processos que garantam a consistência dos dados.

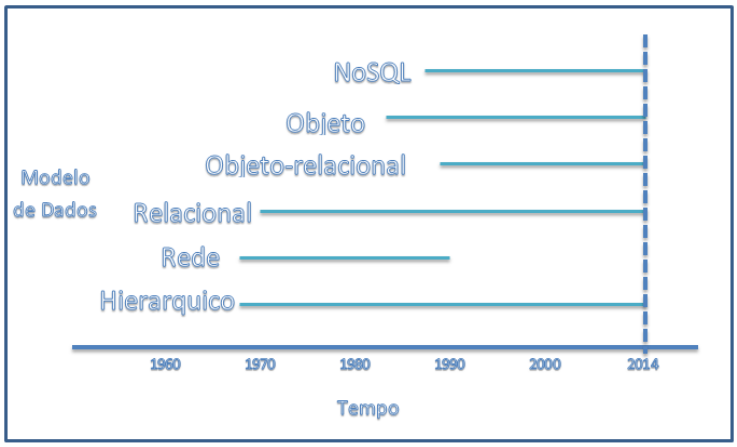
O último V, **valor**, considera que informação é poder, informação é patrimônio. A combinação "volume + velocidade + variedade + veracidade", além de todo e qualquer outro aspecto que caracteriza uma solução de Big Data, se mostrará inviável se o resultado não trouxer benefícios significativos e que compensem o investimento.

O que acontece agora? Diante destas definições, importantes à implementação de SGBDs que suportem a estratégia definida pelo Big Data podemos sugerir diferentes soluções. Bancos do tipo NoSQL são mais flexíveis, sendo inclusive compatíveis com um grupo de premissas que "compete" com as propriedades ACID dos SGBDs tradicionais: **a BASE** (**BAsically Available, Soft state, Eventually consistency** – Basicamente disponível, estado leve, eventualmente consistente).

Exemplos de bancos de dado NoSQL são o Cassandra, o MongoDB, o HBase, o CouchDB e o Redis. Mas, quando o assunto é Big Data, apenas um banco de dados do tipo não basta. É necessário também contar com ferramentas que permitam o tratamento correto do volume de dados. Neste ponto, o Hadoop é, de longe, a principal referência.

O **Hadoop** é uma plataforma *open source* desenvolvida especialmente para processamento e análise de grandes volumes de dados, sejam eles estruturados ou não estruturados. Pode-se dizer que o projeto teve início em meados de 2003, quando o Google criou um modelo de programação que distribui o processamento a ser realizado entre vários computadores para ajudar o seu mecanismo de busca a ficar mais rápido e livre das necessidades de servidores poderosos (e caros). Esta tecnologia recebeu o nome de **MapReduce**.

Vamos ficando por aqui, isso é o que nos interessa para o contexto histórico. Apresentamos abaixo uma figura com uma evolução dos modelos de dados ao longo do tempo.




Vamos agora fazer uma questão sobre o assunto.

CAIU na prova!

## 7. ANO: 2014 ÓRGÃO: TJ-SE PROVA: ANALISTA JUDICIÁRIO – BANCO DE DADOS

Acerca de bancos de dados semiestruturados e bancos de dados NOSQL, julgue os itens subsecutivos.

[86] Bancos de dados NOSQL orientados a documentos são apropriados para o armazenamento de dados semiestruturados.

[87] Para garantir a eficiência das consultas a bancos de dados semiestruturados, é fundamental a adoção de técnica de indexação que leve em consideração, além das informações, as propriedades estruturais dos dados.

[88] Devido à escalabilidade esperada para os bancos de dados NOSQL, a implementação desses bancos utiliza modelos de armazenamento de dados totalmente distintos dos utilizados em sistemas relacionais.

**Comentário:** Vimos que um dos desafios que os banco de dados NoSQL tenta resolver tem relação com os tipos de dados variados, complexos e/ou semiestruturados. Assim podemos considerar a alternativa 86 como **correta**.

A questão 87 envolve alguns conceitos interessantes. Começa falando sobre dados semiestruturados, por exemplo, XML ou JSON. Consultas em bancos de dados semiestruturados consideram tanto a estrutura quanto os valores. Outra questão é a criação de índice sobre um conjunto de dados semiestruturados. Para avaliar se um índice deve ou não ser criado é importante usar as informações sobre a estrutura dos dados e os valores armazenados. Neste caso, considerando a necessidade de um espaço maior para armazenamento e do custo de manutenção, a criação do índice deve melhorar a performance para ser de fato implementado. Logo, a assertiva está **correta**.

A alternativa 88 vai exigir conhecimento sobre os modelos de armazenamento utilizados por bancos de dados NoSQL. Quando tratamos de bases de dados NoSQL podemos classifica-las em quatro diferentes tipos, são eles:

**Chave/valor (Key/Value):** conhecidos como tabelas de hash distribuídas. Armazenam objetos indexados por chaves, e facilita a busca por esses objetos a partir de suas chaves.

**Orientados a Documentos:** os documentos dos bancos são coleções de atributos e valores onde um atributo pode ser multivalorado. Em geral, os bancos de dados orientados a documento não possuem esquema, ou seja, os documentos armazenados não precisam possuir uma estrutura em comum. Essa característica faz deles boas opções para o armazenamento de dados semiestruturados.

**Colunar:** Bancos relacionais normalmente guardam os registros das tabelas contiguamente no disco. Por exemplo, caso se queira guardar id, nome e endereço de usuários em um banco de dados relacional, os registros seriam:

Id1, Nome1, Endereço1;

Id2, Nome2, Endereço2.

Essa estrutura torna a escrita muito rápida, pois todos os dados de um registro são colocados no disco com uma única escrita no banco. Também é eficiente caso se queira ler registros inteiros. Mas para situações onde se quer ler algumas poucas colunas de muitos registros, essa estrutura é pouco eficiente, pois muitos blocos do disco terão de ser lidos.

Para esses casos onde se quer otimizar a leitura de dados estruturados, bancos de dados de famílias de colunas são mais interessantes, pois eles guardam os dados contiguamente por coluna.

O exemplo anterior em um banco de dados dessa categoria ficaria:

Id1, Id2; Nome1, Nome2; Endereço1, Endereço2.

Os bancos de dados de famílias de colunas são mais interessantes para processamento analítico online (OLAP). ***Bigtable*** é uma implementação da Google dessa categoria de bancos de dados.

**Orientado a Grafos:** diferente de outros bancos de dados NoSQL, esse está diretamente relacionado a um modelo de dados estabelecido, o modelo de grafos. A ideia desse modelo é representar os dados e/ou o esquema dos dados como grafos dirigidos, ou como estruturas que generalizem a noção de grafos. O modelo de grafos é aplicável quando “informações sobre a interconectividade ou a topologia dos dados são mais importantes, ou tão importante quanto os dados propriamente ditos”. Possui três componentes básicos: os nós (são os vértices do grafo), os relacionamentos (são as arestas) e as propriedades (ou atributos) dos nós e relacionamentos.

Agora vamos voltar a questão, precisamos responder a seguinte pergunta: nenhum dos modelos acima tem relação com bancos de dados relacionais? Ou ainda, não é possível criar estruturas em SGBDs relacionais que representem esses conceitos? Veja a figura abaixo:




**Modelo de Dados Hierárquicos**
Primeiro modelo de dados a ser reconhecido. Usa uma estrutura de árvores onde cada registo é considerado uma coleção de campos ou atributos.

**Modelo de Dados Relacional**
Sucessor do modelo Hierárquico. Baseia-se no conceito de Entidades e Relacionamentos.

**Melhorias nos SGBD's**
Os Sistemas Gerenciadores de Banco de Dados começam a ser melhorados devido a grande aceitação dos usuários.

**Modelo de Dados NoSQL**
Surgem as primeiras alternativas aos modelos relacionais baseados em documentos, chave-valor ou famílias de colunas.

**Modelo de Dados NoSQL**
As bases de dados NoSQL começam a ser reconhecidas devido ao alto poder de performance e escalabilidade.

Para finalizar vamos fazer apenas um comentário sobre escalabilidade: A escalabilidade em um banco de dados relacional pode ocorrer de duas formas: horizontal e vertical. A forma horizontal ocorre pela utilização de mais equipamentos e particiona a estrutura de dados de acordo com critérios estabelecidos. A forma vertical ocorre pelo aumento da capacidade do equipamento em que o sistema gerenciador de banco de dados está instalado. Bases de dados NoSQL têm como um de seus motivadores o baixo custo para realizar uma escalabilidade horizontal, o que torna possível o uso de equipamentos mais acessíveis. Além disso, proporciona um modelo de particionamento nativo (Sharding).

**Gabarito: C C E.**

---

Vejamos agora uma questão sobre o assunto.




**8. ANO: 2017 ÓRGÃO: TST PROVA: TÉCNICO JUDICIÁRIO – PROGRAMAÇÃO**

[57] Considere que um determinado tipo de banco de dados organiza os dados na forma de uma pirâmide, onde o registro principal no topo da pirâmide é chamado registro raiz. Os registros são organizados como pai e filho onde um registro filho sempre tem apenas um registro pai ao qual ele está ligado, como em uma árvore familiar normal. Em contrapartida, um registro pai pode ter mais de um registro filho a ele ligado.

Trata-se do Banco de Dados

a) hierárquico.

b) relacional.

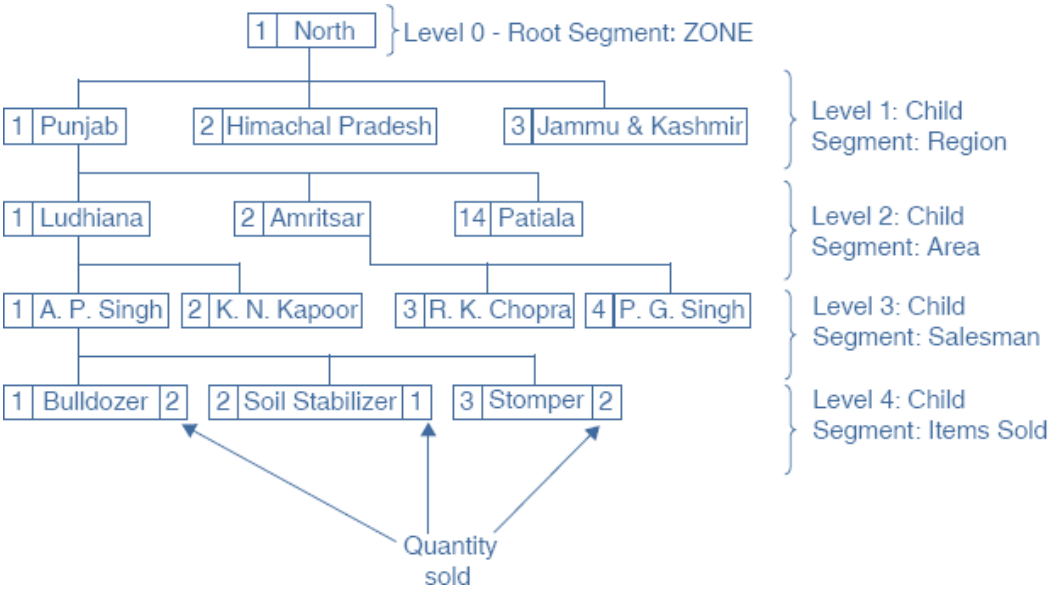
c) orientado a objeto.

d) objeto-relacional.

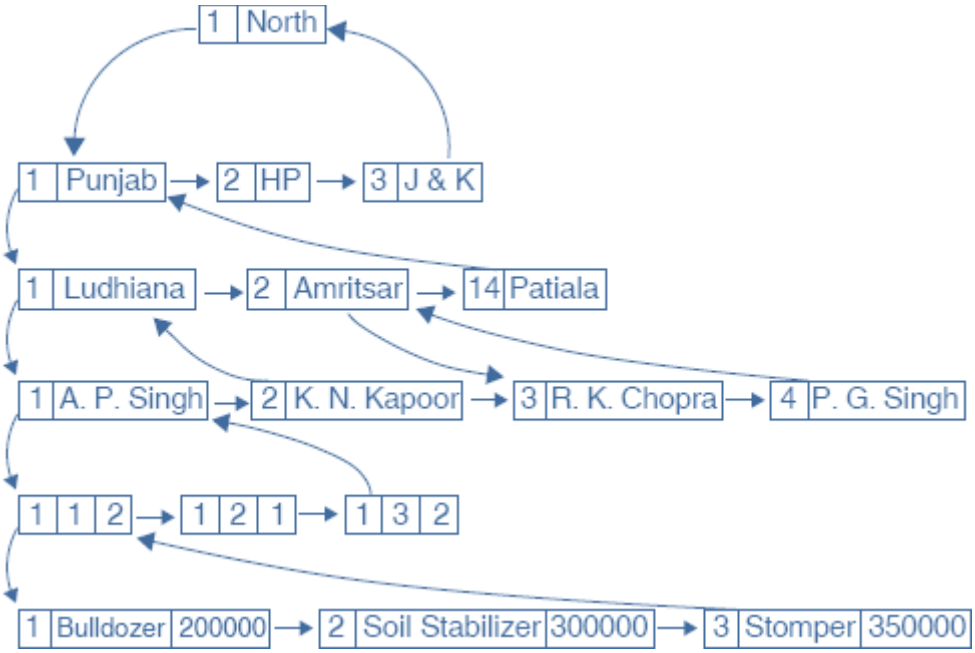
e) de rede.

**Comentário:** Vamos fazer uma rápida revisão sobre os conceitos associados aos modelos de banco de dados orientados a registros:

O **modelo hierárquico** de dados foi o primeiro modelo de dados desenvolvido na década de 1960. O modelo de dados hierárquico foi chamado de IMS (Information Management System) e foi desenvolvido pela IBM e pela Rockwell Company. Ele foi amplamente utilizado durante os anos 1960 e 1970. As entidades e as relações entre entidades foram estruturadas e gerenciadas com a ajuda de uma **estrutura em forma de árvore**. Nesta árvore, existe uma **raiz** que está relacionada aos seus **filhos**. Uma raiz é conhecida como pai. Podemos observar que essa é a nossa resposta. Veja a figura abaixo com a representação de uma estrutura hierárquica.




Já o **modelo de dados de rede** representa dados usando o **link** entre os registros. O registro pai é chamado de **Registro Proprietário**, e o registro filho é chamada de **Registro de Membro**. Se os registros Proprietário e Membro estiverem relacionados com o relacionamento muitos-para-muitos, eles estão conectados através do registro do conector que é conhecido como **Set**. Agora vejamos um modelo semelhante ao visto acima, agora com as características do modelo em rede:




Temos ainda o conceito de **modelo relacional** foi dado por E. F. Codd, em 1970, em seu documento histórico sobre o modelo de dados relacionais. No modelo relacional, os dados são representados em uma forma tabular denominada, **relação** (tabela), e estão associados a relacionamentos. Portanto, o nome desse modelo é o modelo de dados relacional. Cada **entidade** é convertida em relação e a associação é tratada através de **chaves primárias e estrangeiras**. Cada ocorrência da entidade é conhecida como **tupla** (registro) e a característica de uma entidade é chamada de **atributo** (coluna). É muito fácil representar a relação muitos-para-muitos usando o modelo de dados relacionais. O modelo relacional é amplamente utilizado em todo o mundo, hoje em dia, para armazenar dados. Vejamos agora um exemplo de um banco de dados relacional de uma livraria on-line.

**BOOK**

| ISBN | Book_title | Category | Price | Copyright_date | Year | Page_count | P_ID |
|---|---|---|---|---|---|---|---|
| 001-354-921-1 | Ransack | Novel | 22 | 2005 | 2006 | 200 | P001 |
| 001-987-760-9 | C++ | Textbook | 25 | 2004 | 2005 | 800 | P001 |

**PUBLISHER**

| P_ID | Pname | Address | State | Phone | Email_ID |
|---|---|---|---|---|---|
| P001 | Hills Publications | 12, Park street, Atlanta | Georgia | 7134019 | H_pub@hills.com |

**REVIEW**

| R_ID | ISBN | Rating |
|---|---|---|
| A002 | 001-987-760-9 | 6.0 |
| A006 | 001-354-921-1 | 7.5 |
| A008 | 001-987-760-9 | 7.2 |

**Gabarito: A**

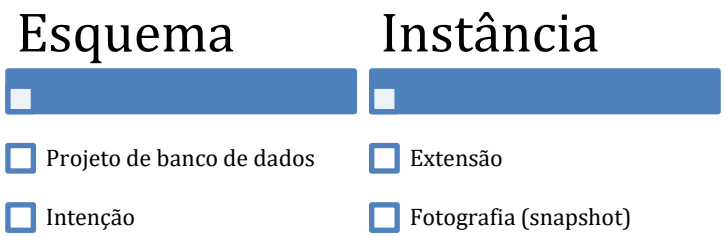
# MODELO DE DADOS E ARQUITETURA EM TRÊS ESQUEMAS

Na parte anterior do nosso curso, tratamos das definições relacionadas aos termos básicos SGBD, BD, SBD e dicionário de dados. Agora vamos tratar de outro grupo de conceitos. Este grupo está relacionado ao entendimento dos diversos participantes do processo de definição de um banco de dados, para definir os diferentes **níveis de modelos de dados** que dão origens aos esquemas. A partir desses esquemas podemos obter instâncias.

## INSTÂNCIAS X ESQUEMAS

Antes de adentrar no estudo dos modelos de dados queria que você entendesse a diferença entre esquema e instância. O **esquema** é a definição da estrutura do modelo e a **instância** se refere aos dados armazenados em um esquema em um momento do tempo.

Para descrever os modelos utilizamos os elementos fornecidos por cada um deles e construímos esquemas. Conhecido com o projeto geral do BD é basicamente a descrição do banco, conhecido também como a **intenção** ou **conotação**. Baseado nesse esquema é possível criar uma instância do BD. Essa coleção de dados armazenados no banco de dados em um determinado instante é o próprio banco de dados, também denominada de **extensão**.




## MODELO DE DADOS

Um **modelo de dados** fornece um significado necessário para permitir a **abstração dos dados**, ocultando detalhes de armazenamento. Pode ser visto como uma coleção de conceitos que são usados para descrever a estrutura de um banco de dados. Cada modelo deve definir uma coleção de ferramentas conceituais para as seguintes tarefas:

**(1) descrição de dados**
**(2) relacionamentos entre eles**
**(3) a semântica dos dados**
**(4) restrições de consistência.**

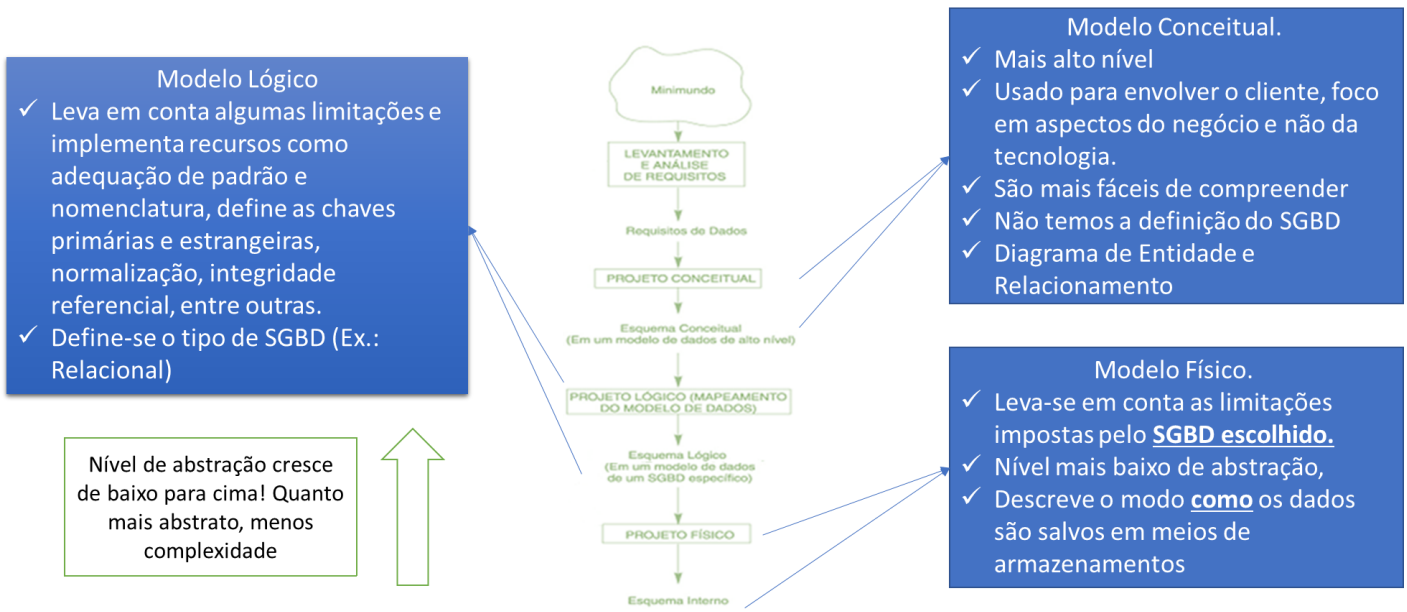
Boa parte dos modelos também dá **suporte a operações**, algumas dessas operações podem representar o aspecto dinâmico ou comportamento de uma aplicação de banco de dados.

Os **modelos de dados** podem ser divididos em três categorias de acordo com os tipos de conceitos usados para descrever a estrutura do banco de dados. No nível mais alto temos os modelos de dados

**conceituais** que apresentam os dados da forma como os usuários finais percebem. Em um nível intermediário está a classe de modelos de dados **representacionais** (ou de implementação) que fornece um entendimento aos envolvidos no processo de desenvolvimento do BD, mas já introduz informações sobre a forma pela qual os dados são armazenados dentro de um computador. O último nível apresenta os modelos de dados **físicos** ou de baixo nível. Observem na figura a seguir:




Antes de continuarmos gostaria de apresentar para você uma figura que mostra a existência de uma evolução ou um refinamento nos modelos de dados. Esse refinamento reduz o nível de abstração permitindo a implementação da estrutura de dados no disco rígido ou em outro dispositivo físico. O passo a passo do projeto de banco de dados será visto em outro momento neste curso.

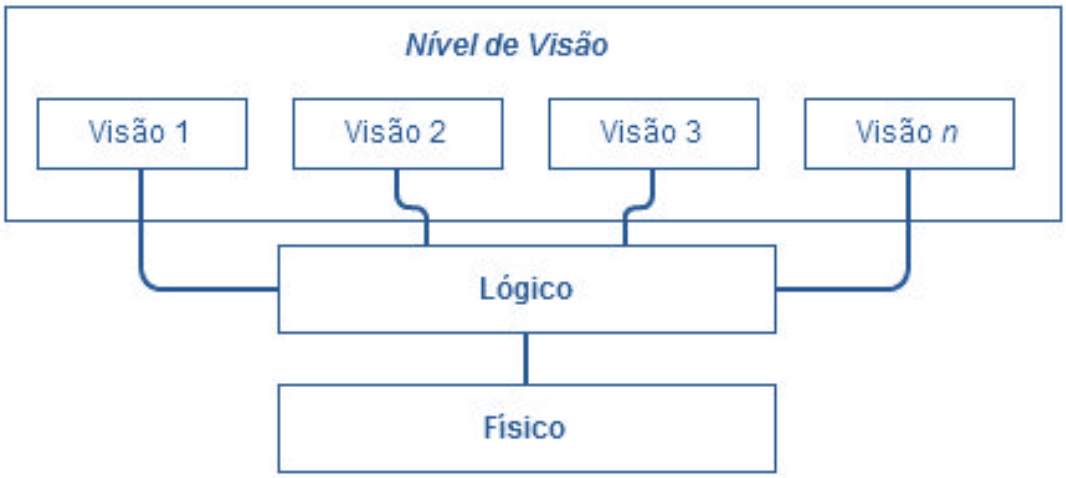



Após conhecermos a divisão de modelo de dados, vamos mudar nosso foco para outra classificação presente no contexto de banco de dados. Silberschatz apresenta um conceito de níveis de abstração. O nível de abstração **mais baixo** ou **físico** descreve **como** os dados realmente são armazenados. Este nível descreve em detalhes estruturas de dados complexas.

O próximo nível de abstração descreve **que** dados estão armazenados no banco de dados e que relações existem entre eles. O **nível lógico** descreve o banco de dados inteiro em termos de um pequeno número de estruturas relativamente simples. Embora a implementação das estruturas simples no nível lógico possa envolver estruturas complexas em nível físico, o usuário do nível lógico não precisa tomar ciência desta complexidade.

O nível de abstração mais alto descreve apenas parte do banco de dados. Muitos usuários de sistema de banco de dos não precisam de todas as informações armazenadas. Em vez disso, eles precisam apenas de uma parte do banco de dados. O **nível de visão** (*view*) existe para simplificar sua interação com o sistema, que pode fornecer muitas visões para o mesmo banco de dados.

A hierarquia de níveis de abstração de dados pode ser vista na figura abaixo:




## ARQUITETURA TRÊS ESQUEMAS

Agora que já entendemos de modelos de dados, instâncias e esquemas, vamos passar para a arquitetura em três esquemas. O American National Standards Institute (ANSI) através do Standards Planning and Requirements Committee (SPARC) estabeleceu um padrão para o desenvolvimento de tecnologias de base de dados, definindo uma arquitetura de três níveis independentes: interno, conceitual e externo.

Essa arquitetura tem por objetivo separar o usuário da aplicação do banco de dados físico. Possuem, logicamente, os esquemas definidos em três níveis distintos:

◉ Nível interno - (também conhecido como nível de armazenamento) é o mais próximo do meio de armazenamento físico - ou seja, é aquele que se ocupa do modo como os dados são fisicamente armazenados dentro do sistema.
◉ Nível conceitual - (também conhecido como nível lógico de comunidade, ou às vezes apenas nível lógico, sem qualificação) é um nível "indireto" entre os outros dois.

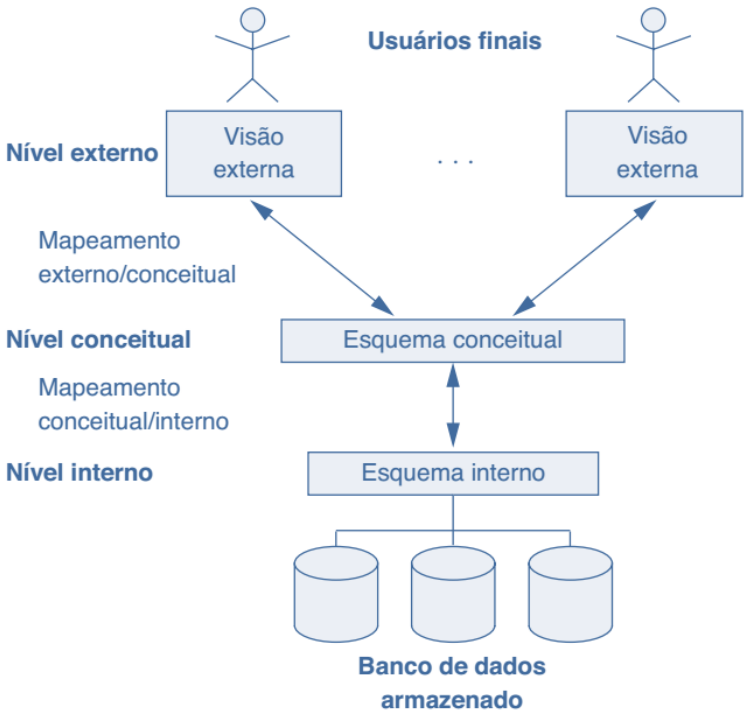
◉ Nível externo ou visão - (também conhecido como nível lógico do usuário) é o mais próximo dos usuários – ou seja, é aquele que se ocupa do modo como os dados são vistos por usuários individuais.

Podemos usar partir da arquitetura de três esquemas para explicar o último assunto desta nossa parte da aula. A **independência dos dados** que nada mais é do que a capacidade de alterar o esquema em um nível dos sistemas de banco de dados sem alterar o esquema no nível mais alto ou, em outras palavras a habilidade de modificar a definição de um esquema em um nível sem afetar a definição do esquema em um nível mais alto.

Segundo Navathe é possível definir dois tipos de independência de dados:

1. **Independência lógica de dados** - a capacidade de alterar o esquema conceitual sem ter de alterar os esquemas externos ou os aplicativos.
2. **Independência física de dados** - a capacidade de alterar o esquema interno sem ter de alterar o esquema conceitual.

Apresentamos a seguir uma figura que descreve os níveis da arquitetura em três esquemas. Observem que pela definição de independência de dados é necessário a existência de um nível superior ao esquema alterado. Desta forma só temos o conceito para os níveis conceitual e interno da figura.




De uma forma simples, cada um dos níveis possui uma função dentro das suas respectivas abstrações. O nível de visão do usuário determina a parte em que o usuário tem acesso. O nível conceitual identifica os dados armazenados e suas relações. Por fim, o nível físico é o nível mais baixo de abstração, define a maneira como os dados estão armazenados. Vejamos mais algumas questões sobre o assunto:




**9. Ano: 2016 Órgão: TRE-PI Prova: Analista Judiciário - Análise de Sistemas**

A respeito das características de um SGBD e das atividades de administração de banco de dados, assinale a opção correta.

a) Para fins práticos, é necessário distinguir diferentes cardinalidades máximas, que podem ser maiores ou iguais a zero.

b) A característica autodescritiva de um banco de dados define que o banco de dados contém o próprio dado assim como uma descrição desses dados e suas restrições. Essas descrições e restrições estão armazenadas no catálogo (dicionário) do SGBD.

c) A independência física de dados consiste na habilidade de modificar o esquema conceitual sem a necessidade de reescrever os programas aplicativos. As modificações no nível conceitual são necessárias quando a estrutura lógica do banco de dados é alterada.

d) Na linguagem SQL, os comandos DDL GRANT e ROLLBACK permitem a implementação de um controle de acesso discricionário, criando e retirando permissões no banco de dados.

e) A coleção das informações armazenadas em um banco de dados, em determinado momento, corresponde ao esquema do banco de dados.

**Comentário:** Observe que a descrição presente na alternativa B está perfeitamente adequada. Tente encontrar os erros das demais alternativas. Apenas por curiosidade, na alternativa D, a linguagem SQL possui algumas subdivisões dos comandos. Os comandos GRANT e REVOKE server para implementação do controle de acesso discricionário. O ROLLBACK, juntamente com o COMMIT e SAVEPOINT servem para controle de transações.

Outro ponto importante, é que, na alternativa A, a cardinalidade máxima tem que ser obrigatoriamente maior ou igual a 1. A justificativa para isso será dada na próxima aula.

**Gabarito: B.**

---




**10. Ano: 2014 Órgão: TJ-CE Prova: Analista Judiciário - Ciências Computação**

Considerando o sistema gerenciador de banco de dados (SGBD), assinale a opção correta acerca de bancos de dados.

a) Enquanto a DDL (Data Definition Language) é utilizada para definir a estrutura do banco de dados, a SDL (Storage Definition Language) é utilizada para especificar o esquema conceitual e seus mapeamentos com o esquema interno.

b) A informação armazenada no catálogo do SGBD é denominada metamodelo.

c) Na independência de dados do programa, propriedade do SGBD, a estrutura dos arquivos de dados é armazenada no catálogo separadamente dos programas de acesso.

d) Na arquitetura de três esquemas de um banco de dados, o nível conceitual é responsável por descrever de forma detalhada as estruturas de armazenamento físico, incluindo os relacionamentos entre as tabelas.

e) Na arquitetura de três esquemas, a capacidade de alterar o esquema interno sem ter de alterar o esquema conceitual consiste na independência lógica de dados.

**Comentário:** Nesta questão, tão importante quanto saber que a alternativa C está correta, e esse assunto nós já vimos no início da aula, é entender porque as outras estão erradas. Perceba primeiramente que, se formos teoricamente precisos deveríamos trocas SGBD por SDB. (lembra?!, SGBD + DB = SBD) A questão é que alguns autores falam da abordagem de sistemas de gerenciamento de banco de dados. Assim, essa acaba sendo a menos errada das alternativas.

Na alternativa A, o examinador sugere que SDL, é utilizada para descrever um esquema conceitual. Vejam que SDL está relacionada com armazenamento, e, quando existe de forma separada em uma linguagem, é utilizada para definir os detalhes do nível interno.

Na alternativa B, no lugar de meta modelo, o correto seria metadados.

Na alternativa D, o nível descrito é o nível interno.

Por fim, a alternativa E trata de independência física e não lógica como descrito no texto.

**Gabarito: C.**

---




**11. Ano: 2016 Órgão: TCE-PA Prova: Auditor de Controle Externo - Área Informática - Analista de Sistema**

Julgue o item subsequente, no que se refere a sistemas de gerenciamento de bancos de dados (SGBD).

Independência lógica de dados refere-se à capacidade de alterar o esquema conceitual sem a necessidade de alterar os esquemas externos ou os programas de aplicação.

**Comentário:** Lembre-se que a independência lógica ou conceitual atual no nível intermediário da arquitetura em 3 esquemas. Ela está relacionada ao fato de que modificações nesta camada da arquitetura não gera necessidade de alteração nos esquemas externos ou programas de aplicação. Desta forma, a alternativa encontra-se correta.

**Gabarito: C.**

---



**12. Ano: 2016 Órgão: TCE-PA Prova: Auditor de Controle Externo - Área Informática - Administrador de Banco de Dados**

Com relação a sistemas gerenciadores de bancos de dados (SGBD), julgue o próximo item.

No nível conceitual da arquitetura de três camadas de banco de dados, cada esquema externo descreve a parte do banco que interessa a determinado grupo de usuários e oculta desse grupo o restante do banco de dados.

**Comentário:** Perceba que essa questão apresenta uma casca de banana das mais malvadas. Ele mistura o nível conceitual da arquitetura em três esquemas com o nível externo. Sabemos que os esquemas externos estão associados as diferentes visões dos grupos de usuários. Tal fato reflete os interesses distintos de cada grupo quanto ao acesso as informações do banco de dados. Assim, podemos marcar nosso gabarito como errado.

**Gabarito: E.**

---




**13. ANO: 2010 ÓRGÃO: TCE-SP PROVA: AGENTE DA FISCALIZAÇÃO FINANCEIRA - CONHECIMENTOS BÁSICOS**

As três visões da arquitetura básica de um SGBD, pela ordem, desde a mais próxima do usuário até a mais distante, são:

A externa, conceitual e interna.

B externa, interna e conceitual.

C conceitual, interna e externa.

D conceitual, externa e interna.

E interna, conceitual e externa.

**Comentário.** Ao analisar as alternativas temos que ter em mente duas informações relevantes de enunciados. Primeiro é solicitado as três visões da arquitetura, desta forma podemos concluir que o examinador está se referindo a arquitetura em três esquemas. A outra informação é que a questão pede para você colocar em ordem crescente, da mais próxima do usuário até a mais distante. Com essas duas informações podemos relembrar dos conceitos vistos anteriormente e descrever a seguinte ordem: **externa, conceitual e interna**.

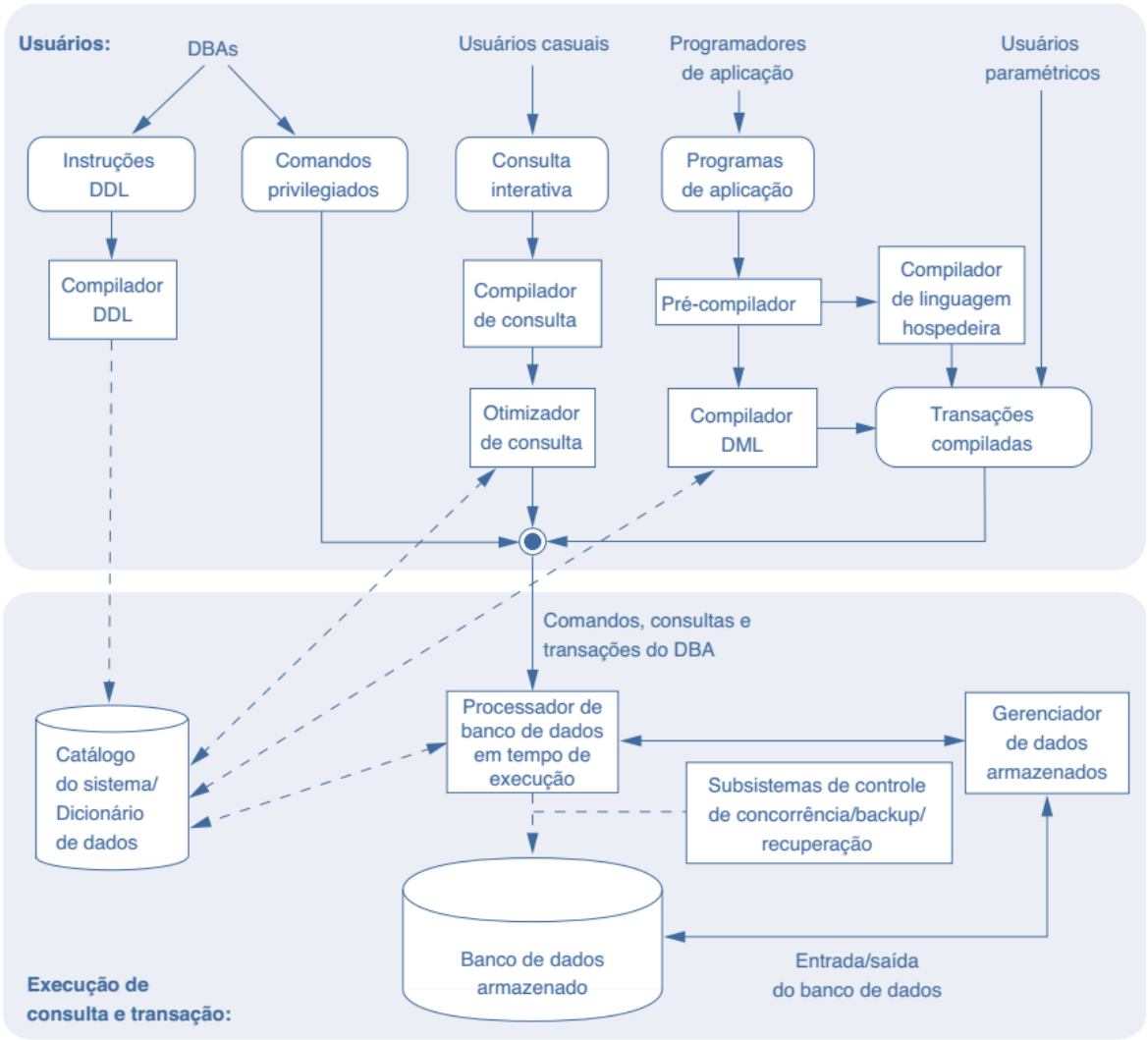
**Gabarito: A.**

---

## AMBIENTE DO SISTEMA DE BANCO DE DADOS

A figura a seguir apresenta, de forma simplificada, os componentes típicos de um SGBD. Perceba que temos uma divisão em duas partes. A parte superior mostra os usuários e a forma de interação deles com o SGBD. Cada tipo de usuário possui a sua interface de relacionamento.

A parte inferior ilustra os detalhes internos do SGBD, eles são responsáveis pelo armazenamento dos dados e processamento das transações dos usuários. Vejam que o **banco de dados** e o **catálogo do SGBD** estão representados por um cilindro, representando o fato de serem armazenados permanentemente. Os **discos rígidos** representam uma das diversas possibilidades de armazenamento não volátil ou permanente dos dados.




Nas próximas linhas vou descrever um pouco dos componentes da figura acima. Os conceitos podem ser úteis para que você entenda um pouco da "caixa preta" por trás de um SGBD.

O acesso aos discos rígidos é controlado pelo sistema operacional ou pelo próprio SGBD. O controle vai escalonar as operações de leitura e escrita sobre o disco. Quando consideramos a execução de uma transação sobre o banco de dados, o intervalo de tempo gasto na transferência de dados entre

a memória e o disco é considerado um gargalo no processamento. Para otimizar o processo alguns SGBDs podem contar com um **módulo de gerenciamento de buffer** que planeja a troca de dados entre a memória principal e o disco.

Outro módulo, o **gerenciador de dados armazenados**, controla o acesso às informações do SGBD que estão armazenadas, seja no catálogo de dados ou no bando de dados. Ele utiliza os serviços básicos do sistema operacional para executar operações de entrada/saída (leitura/escrita) de baixo nível entre o disco e a memória principal.

Observa-se, na parte superior da figura, as interfaces para os diferentes usuários do sistema. De um lado temos **os usuários casuais** que trabalham com interfaces interativas para formular consultas. Logo em seguida, visualizamos os **programadores de aplicação** que usam uma linguagem de programação hospedeira para ter acesso aos dados. Por fim, temos os **usuários paramétricos** que inserem valores para os parâmetros predefinidos pelas transações.

No parágrafo anterior, falamos apenas das manipulações de dados que podem ser feitas sobre as bases de dados. Essas operações incluem consultas, inserções, deleções e atualizações das informações armazenadas. Contudo, é necessário, antes de manipularmos os dados, construir as estruturas do banco de dados. A linguagem de definição de dados ou *data definition language* é utilizada pelo **DBA** para descrever os objetos presentes na base de dados. Uma tabela do modelo relacional é um exemplo de objeto que podemos criar por meio de uma **instrução DDL** em nossa base de dados.

Quando o DBA digita uma instrução ou comando DDL, essa é enviada ao **compilador DDL.** Um compilador transforma o código fonte (da linguagem de programação) em um código em que o computador entenda. O compilador da DDL processa as definições de esquema especificadas e armazena as descrições de esquema (metadados) no catálogo do SGBD. Esse fluxo pode ser observado no lado esquerdo da figura anterior.

Outra função do DBA é o ajuste fino ou *tuning* do sistema de gerenciamento de banco de dados, bem como a configuração de parâmetros que são feitos por meio dos **comandos privilegiados**. Apenas para exemplificar, um comando presente na maioria dos SGBDs é o *REORG*, serve para reorganizar uma tabela ou índice na estrutura física do banco de dados.

Os usuários casuais interagem usando alguma interface de consulta interativa. Essas consultas são analisadas e validadas pela exatidão da sintaxe da consulta, os nomes de arquivos e elementos de dados, e assim por diante, por um **compilador de consulta**. Essa consulta interna está sujeita a melhorias feitas pelo otimizador de consultas, que se preocupa com o rearranjo e a possível reordenação de operações, com a eliminação de redundâncias e uso dos algoritmos e índices corretos durante a execução.

Ele consulta o catálogo do sistema em busca de informações estatísticas e outras informações físicas sobre os dados armazenados, gerando um código executável. Este por sua vez realiza as operações necessárias para a consulta e faz chamadas ao **processador em tempo de execução** (falaremos mais sobre ele logo mais).

Os programadores de aplicação escrevem programas em linguagens hospedeiras, como Java e C#, que são submetidas a um **pré-compilador**. Este extrai os comandos DML do programa de aplicação.

Para entender melhor como funciona essa divisão vamos partir para um exemplo prático. No exemplo abaixo temos um código Java com um comando SQL. O comentário no código delimita o início da instrução SQL dentro do código Java.

```
public boolean verificarUsuario(String login, String senha){
    String sql = "";
    Connection conn = conectarBD();
    //INSTRUÇÃO SQL
    sql += "select nome from usuarios ";
    sql +="where login = " + "'" + login + "'";
    sql += " and senha = " + "'" + senha + "';";
    try{
        Statement st = conn.createStatement();
        ResultSet rs = st.executeQuery(sql);
        if(rs.next()){
            result = true;
            nome = rs.getString("nome");}
    }catch (Exception e) {  }
    return result; }
```

Observem que o comando *select*. Ele está escrito na linguagem SQL considerada uma DML. Esses comandos são enviados ao compilador DML para serem compilados em código objeto com acesso ao banco de dados. O restante do programa é enviado ao compilador da linguagem hospedeira. Os códigos objetos para os comandos DML e o restante do programa são ligados (‘linkados’) formando uma transação programada ou compilada.

As transações programadas são executadas repetidas vezes pelos usuários paramétricos, que apenas fornecem os parâmetros para as transações. No nosso exemplo anterior, as informações necessárias são *login* e senha. Cada execução é considerada uma transação separada. Outro exemplo de transação é o saque no caixa eletrônico, no qual o número da conta e o valor são fornecidos como parâmetros.

Na parte inferior da figura temos o **processador de banco de dados em tempo de execução (PBDTE).** Ele é responsável por executar os comandos privilegiados, os planos de consulta executáveis e as transações programadas. Para isso são utilizadas informações e dados estatísticos do catálogo do sistema. O PBDTE também trabalho com o **gerenciador de dados armazenados**.

Os sistemas de controle de concorrência, backup e recuperação são apresentados como um módulo da figura. Eles são integrados ao processador de banco de dados em tempo de execução para fins de gerenciamento de transações. Você precisa ter em mente que esses controles são necessários para o perfeito funcionamento do SGBD. O backup é utilizado durante a recuperação caso alguma falha aconteça. A concorrência entre transações deve existir dentro de limites bem definidos para evitar que o banco de dados entre em um estado inconsistente.

A figura apresentada não pretende descrever um SGBD específico nem esgotar suas funcionalidades. Nossa ideia foi ilustrar os módulos básicos de um SGBD e estruturar seu raciocínio. Lembre-se que um SGBD é um sistema informatizado. Para executar todas as suas tarefas sua implementação é dividida em diferentes módulos. O SGBD interage ainda com o sistema operacional quando o acesso ao disco rígido é necessário. Vamos fazer uma questão que trata do assunto.




**14. Ano: 2010 Órgão: BADESC Cargo: Analista de Sistemas**

Os objetivos dos compiladores DDL, DML e DCL são, respectivamente:

A) criar os objetos do banco de dados, manipular (recuperação, inserção, remoção e alteração) de dados nos objetos criados pela DDL e fornecer privilégio de acesso às informações.

B) fornecer privilégio de acesso às informações, criar os objetos do banco de dados e manipular (recuperação, inserção, remoção e alteração) de dados nos objetos criados pela DDL.

C) manipular (recuperação, inserção, remoção e alteração) de dados nos objetos criados pela DML, criar os objetos do banco de dados e fornecer privilégio de acesso às informações.

D) fornecer privilégio de acesso às informações, manipular (recuperação, inserção, remoção e alteração) de dados nos objetos criados pela DDL e criar os objetos do banco de dados.

E) criar os objetos do banco de dados, fornecer privilégio de acesso às informações e manipular (recuperação, inserção, remoção e alteração) de dados nos objetos criados pela DDL.

**Comentário:** Vimos que o compilador DDL permite que o DBA emita comandos para a criação dos objetos do banco de dados, esses vão fornecer informações sobre as estruturas das tabelas como atributos e restrições. O compilador DML é responsável por transformar o código SQL para manipulação dos dados armazenados.

Por fim, temos o ***Data Control Language – DCL,*** essa parte da linguagem SQL vai permitir aos administradores de banco de dados a distribuição de privilégios de acesso sobre a base de dados. Com essa informação, podemos marcar tranquilamente a resposta na alternativa A.

**Gabarito: A.**

---

Espero que você tenha entendido nossa proposta de apresentar o ambiente do SGBD. Acabamos aqui o nosso primeiro conjunto de assuntos teóricos envolvidos na introdução dos sistemas de bancos de dados.

# RESUMO

Antes de finalizar de fato a nossa aula vamos revisar alguns tópicos importantes que forma vistos que gostaria que você fixasse. Vamos começar com uma citação ao Silberchatz.

> *Um **sistema de gerenciamento de banco de dados (SGBD)** é uma coleção de dados inter-relacionados e um conjunto de programas para acessar esses dados. A **coleção de dados**, normalmente conhecida como **banco de dados**, contém informações relevantes para uma empresa. O **principal objetivo** de um SGBD é proporcionar uma forma de armazenar e recuperar informações de um banco de dados de maneira **conveniente e eficiente**.*

Outro ponto importante são os conceitos hierárquicos de banco de dados vistos ao longo da aula. Reconhecemos que em determinado momento parece que elas estão tratando do mesmo conceito.

Eu gosto sempre de dizer que os conceitos acabam se complementando. Os **modelos de dados** podem ser associados a forma de representação dos dados. Eles variam de acordo com a percepção das pessoas que interagem com o sistema em cada um dos níveis. Sendo assim, temos o **modelo conceitual**, que representa os dados para os usuários de negócio; o **modelo representativo ou lógico**, que estrutura os dados para implementação em um SGBD; e os **modelos físicos,** que descrevem a forma como os dados são armazenados nos dispositivos de armazenamento, por exemplo, disco rígido.

Por outro lado, temos as hierarquias definidas pelos **níveis de abstração** e pela **arquitetura em 3 esquemas**. Uma característica importante em ambas é que no nível mais alto de abstração elas separam o banco de dados de acordo com grupos de usuários. Imagine que cada setor de uma empresa tenha acesso a apenas uma parte dos dados. O **nível de visão** ou a **visão externa** descreve exatamente essa perspectiva.

Abaixo deste nível temos o **nível de lógico** na **hierarquia de abstração** e o **nível conceitual** na **arquitetura em três esquemas**. Você deve estar se perguntando como esses níveis se juntam com os modelos de dados? Os modelos de dados conceituais dão origem à modelos de dados lógicos que servem para representar tanto esquemas externos quanto esquemas conceituais da arquitetura em 3 esquemas. É interessante dizer que alguns autores chamam os modelos de dados conceituais de modelos lógicos baseados em objetos e os modelos de dados representativos de modelos lógicos baseados em registros.

Por fim, temos o nível físico ou interno. Esse é igual em todas as hierarquias. Este nível é responsável pela forma de organização dos dados no disco e por descrever os métodos de acesso para o mesmo.

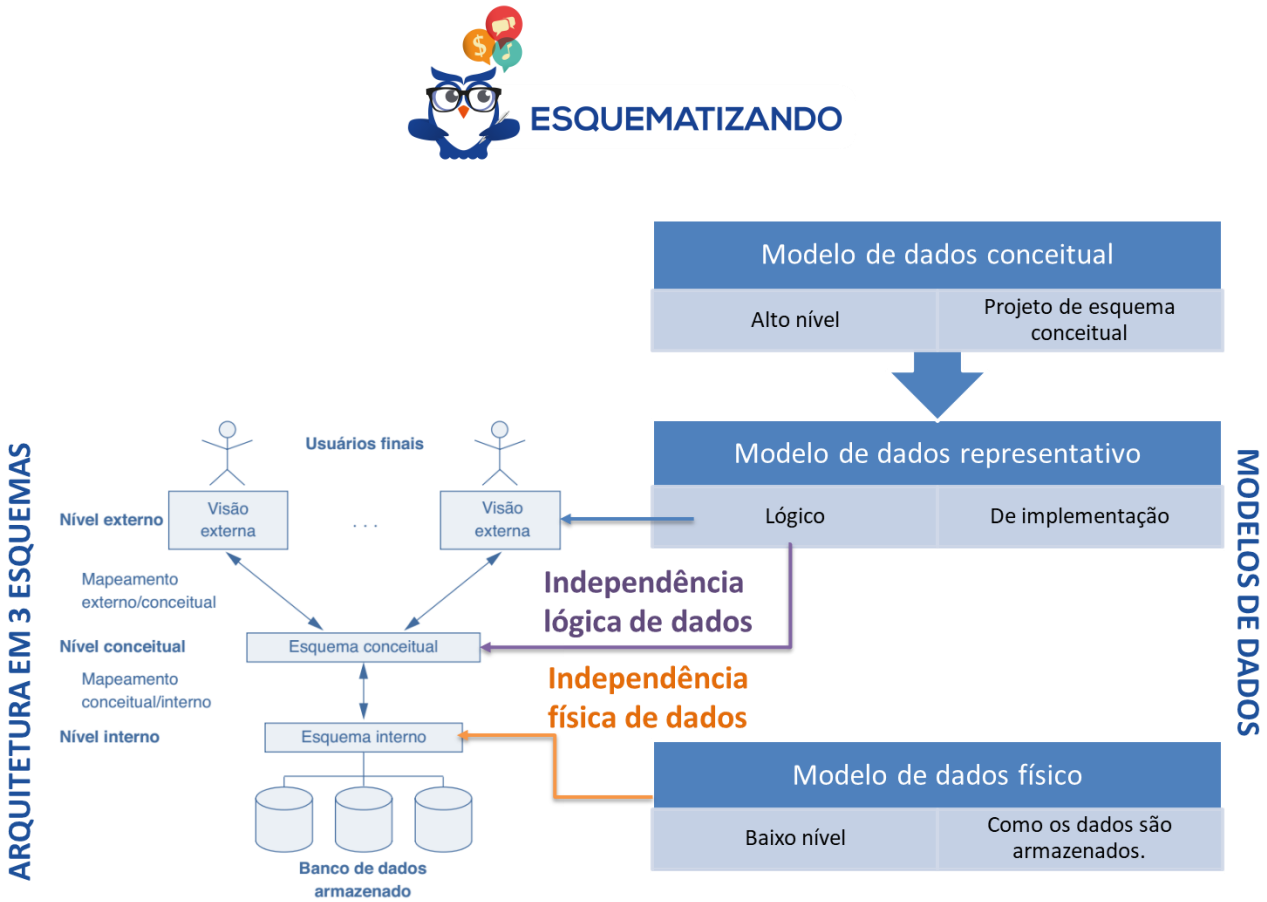
Para finalizar temos que lembrar do conceito de independência de dados que trata da capacidade de alterar o esquema em um nível do sistema de banco de dados sem ter que alterar o esquema nos outros níveis. A independência de dados lógicos é a capacidade de alterar o esquema conceitual sem afetar os esquemas externos ou os programas de aplicativos. O esquema conceitual pode ser alterado devido a mudanças nas restrições ou adição de novos itens de dados ou a remoção de itens de dados existentes.

Já a independência de dados físicos é a capacidade de alterar o esquema interno sem afetar o esquema conceitual ou externo. Um esquema interno pode ser alterado por vários motivos, como por exemplo, para criar uma estrutura de acesso adicional, alterar a estrutura de armazenamento,

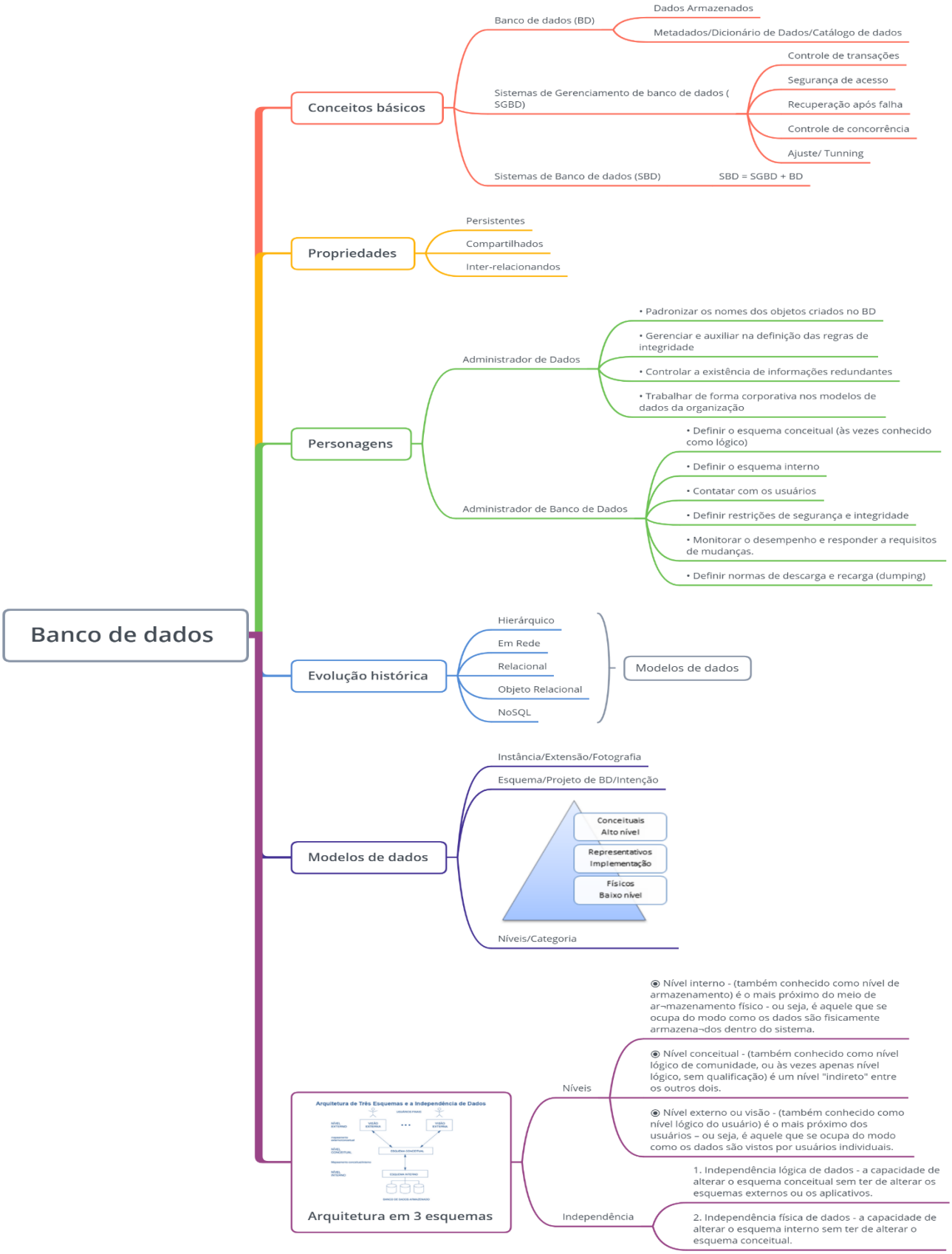
etc. A separação do esquema interno do esquema conceitual facilita a independência física dos dados.

Vejamos então as listas de tópicos e a figura com o relacionamento entre elas:

**Modelo de dados**: conceitual (alto nível), representativos (lógico ou de implementação) e físicos
**Níveis de abstração de dados**: nível de view, nível lógico e nível físico.
**Arquitetura 3 esquemas**: visão externa, esquema conceitual e esquema interno.
**Arquitetura 3 esquemas (níveis)**: nível externo, nível conceitual e nível interno.
**Independência de dados:** lógica e física.



# MAPA MENTAL


Banco de dados
Conceitos básicos
Banco de dados (BD)
Dados Armazenados
Metadados/Dicionário de Dados/Catálogo de dados
Sistemas de Gerenciamento de banco de dados ( SGBD)
Controle de transações
Segurança de acesso
Recuperação após falha
Controle de concorrência
Ajuste/ Tunning
Sistemas de Banco de dados (SBD)
SBD = SGBD + BD
Propriedades
Persistentes
Compartilhados
Inter-relacionandos
Personagens
Administrador de Dados
• Padronizar os nomes dos objetos criados no BD
• Gerenciar e auxiliar na definição das regras de integridade
• Controlar a existência de informações redundantes
• Trabalhar de forma corporativa nos modelos de dados da organização
Administrador de Banco de Dados
• Definir o esquema conceitual (às vezes conhecido como lógico)
• Definir o esquema interno
• Contatar com os usuários
• Definir restrições de segurança e integridade
• Monitorar o desempenho e responder a requisitos de mudanças.
• Definir normas de descarga e recarga (dumping)
Evolução histórica
Hierárquico
Em Rede
Relacional
Objeto Relacional
NoSQL
Modelos de dados
Modelos de dados
Instância/Extensão/Fotografia
Esquema/Projeto de BD/Intenção
Conceituais Alto nível
Representativos Implementação
Físicos Baixo nível
Níveis/Categoria
Arquitetura em 3 esquemas
Níveis
Nível interno - (também conhecido como nível de armazenamento) é o mais próximo do meio de ar¬mazenamento físico - ou seja, é aquele que se ocupa do modo como os dados são fisicamente armazena¬dos dentro do sistema.
Nível conceitual - (também conhecido como nível lógico de comunidade, ou às vezes apenas nível lógico, sem qualificação) é um nível "indireto" entre os outros dois.
Nível externo ou visão - (também conhecido como nível lógico do usuário) é o mais próximo dos usuários – ou seja, é aquele que se ocupa do modo como os dados são vistos por usuários individuais.
Independência
1. Independência lógica de dados - a capacidade de alterar o esquema conceitual sem ter de alterar os esquemas externos ou os aplicativos.
2. Independência física de dados - a capacidade de alterar o esquema interno sem ter de alterar o esquema conceitual.

# QUESTÕES COMENTADAS

## QUESTÕES COMENTADAS

Aproveitamos este espaço para complementar seu entendimento sobre o assunto, sempre colocando **pinceladas extras de conteúdo**. Nosso objeto é criar em você a capacidade de utilizar seu conhecimento, atrelado a algumas heurísticas, para resolver de forma rápida e segura as questões na hora da prova. Alguns assuntos abaixo serão detalhados em aulas subsequentes. Aproveite as questões para ir se **familiarizando com o vocabulário** de banco de dados.




**15. Ano: 2018 Banca: FCC Órgão: DPE-AM Cargo: Analista Área: Banco de Dados Questão: 47**

Uma das propriedades conhecidas em bancos de dados relacionais como ACID é a durabilidade, segundo a qual, em uma transação completada com sucesso, todas as atualizações feitas no banco de dados por essa transação devem persistir

(A) com exceção da ocorrência de falhas na memória do servidor de banco de dados.

(B) mesmo na ocorrência de falhas no sistema de banco de dados, após o término da transação.

(C) a menos que ocorra uma falha no servidor em até 1 segundo, após o término da transação.

(D) exceto se houver outras transações em execução no momento da falha.

(E) com exceção da ocorrência de falhas no barramento de dados do servidor do banco de dados.

**Comentário:** A **durabilidade** é uma propriedade das transações associada ao fato de que, uma vez efetivada a transação não deve ser perdida da base de dados ao menos que outra transação faça isso explicitamente. É importante perceber que, mesmo que aconteça falhas no sistema, ele deve ser capaz de manter o dado armazenado. Sendo assim, podemos marcar nossa resposta na alternativa B. As demais propriedades do acrônimo ACID são: **atomicidade, consistência e isolamento.**

**Gabarito: B**

---




**16. Ano: 2018 Banca: FCC Órgão: DPE-AM Cargo: Analista Área: Banco de Dados Questão: 24**

Dentre os diversos tipos de bancos de dados, existe o denominado bancos de dados objeto-relacionais, que tem como fundamento a

(A) duplicação das tabelas componentes de um banco de dados originalmente relacional.

(B) incorporação de características e recursos da orientação a objetos nos bancos de dados originalmente relacionais.

(C) eliminação do conceito de atributos, existente nos bancos de dados originalmente relacionais.

(D) transformação de todas as tabelas de um banco de dados originalmente relacional em uma única classe da orientação a objetos.

(E) substituição do conceito representado pelas propriedades ACID pelo teorema CAP.

**Comentário:** Os bancos de dados objeto-relacionais tentam fundir as características dos modelos orientados a objetos com os benefícios das estruturas relacionais. Desta forma, podemos marcar nossa resposta na alternativa B. Agora vejamos quais os erros das demais alternativas.

A letra "A" fala de duplicação das tabelas, isso não tem relação com bancos de dados OR. A duplicação controlada de dados está associada a palavra replicação. Já a alternativa "C" fala da eliminação dos atributos, isso não é verdade, o que acontece é que características associadas a OO são incorporadas nos modelos de dados como a herança. A alternativa "D" apresenta um absurdo, pois cada tabela deve ser transformada em uma classe. Por fim, a letra "E" coloca fala que o teorema CAP, que está relacionado a bases de dados NoSQL, substitui o conceito de ACID, que está relacionado às transações do modelo relacional. Logo, todas as demais alternativas estão **erradas**.

**Gabarito: B.**

---




## 17. BANCA: FCC ANO: 2017 ÓRGÃO: TST PROVA: TÉCNICO JUDICIÁRIO – PROGRAMAÇÃO

[57] Considere que um determinado tipo de banco de dados organiza os dados na forma de uma pirâmide, onde o registro principal no topo da pirâmide é chamado registro raiz. Os registros são organizados como pai e filho onde um registro filho sempre tem apenas um registro pai ao qual ele está ligado, como em uma árvore familiar normal. Em contrapartida, um registro pai pode ter mais de um registro filho a ele ligado.

Trata-se do Banco de Dados

a) hierárquico.

b) relacional.

c) orientado a objeto.

d) objeto-relacional.

e) de rede.

**Comentário:** Vamos fazer uma rápida revisão sobre os conceitos associados aos modelos de banco de dados orientados a registros:

O **modelo hierárquico** de dados foi o primeiro modelo de dados desenvolvido na década de 1960. O modelo de dados hierárquico foi chamado de IMS (Information Management System) e foi desenvolvido pela IBM e pela Rockwell Company. Ele foi amplamente utilizado durante os anos 1960 e 1970. As entidades e as relações entre entidades foram estruturadas e gerenciadas com a ajuda de uma **estrutura em forma de árvore**. Nesta árvore, existe uma **raiz** que está relacionada aos seus **filhos**. Uma raiz é conhecida como pai. Podemos observar que essa é a nossa resposta.

Já o **modelo de dados de rede** representa dados usando o **link** entre os registros. O registro pai é chamado de **Registro Proprietário**, e o registro filho é chamada de **Registro de Membro**. Se os registros Proprietário e Membro estiverem relacionados com o relacionamento muitos-para-muitos, eles estão conectados através do registro do conector que é conhecido como **Set**.

Temos ainda o conceito de **modelo relacional** foi dado por E. F. Codd, em 1970, em seu documento histórico sobre o modelo de dados relacionais. No modelo relacional, os dados são representados em uma forma tabular denominada, **relação** (tabela), e estão associados a relacionamentos. Portanto, o nome desse modelo é o modelo de dados relacional. Cada **entidade** é convertida em relação e a associação é tratada através de **chaves primárias e estrangeiras**. Cada ocorrência da entidade é conhecida como **tupla** (registro) e a característica de uma entidade é chamada de **atributo** (coluna). É muito fácil representar a relação muitos-para-muitos usando o modelo de dados relacionais. O modelo relacional é amplamente utilizado em todo o mundo, hoje em dia, para armazenar dados.

**Gabarito: A**

---

CAIU na prova!

**18. BANCA: FCC ANO: 2017 ÓRGÃO: DPE-RS PROVA: ANALISTA – BANCO DE DADOS**

[37] O dicionário de dados de um banco de dados relacional

a) não se aplica a tabelas com pequeno número de registros.

b) não considera o armazenamento da lista de atributos chave das tabelas.

c) armazena, dentre outras informações, nomes de tabelas e de seus atributos.

d) ignora os domínios de cada atributo das tabelas.

e) somente se aplica a tabelas com grande número de atributos.

**Comentário:** O **SGBD** deve fornecer uma função de dicionários de dados. Ele pode ser considerado um banco de dados isolado que contém “os dados sobre os dados”, também chamados de metadados ou descritores. Nele são armazenadas todas as definições dos objetos de banco de dados. Quando tratamos de tabelas do modelo relacional, o dicionário de dados

armazena os nomes das colunas, as restrições de integridade e os tipos de dados, entre outras informações.

Do ponto de vista macro, os dicionários de dados devem armazenar os vários esquemas e mapeamentos e todas as diversas restrições de segurança e integridade. Um dicionário completo também incluirá informações sobre os programas que utilizam determinadas partes do banco de dados, os usuários que exigem certos relatórios, entre outros dados.

Vejam, portanto, que a nossa resposta está presente na alternativa C.

**Gabarito: C**

---




**19. BANCA: FCC ANO: 2016 ÓRGÃO: PREFEITURA DE TERESINA - PI PROVA: TÉCNICO DE NÍVEL SUPERIOR - ANALISTA DE SISTEMAS**

[58] Em relação a projetos de bancos de dados, considere:

I. Tem dependência com a classe do Gerenciador de Banco de Dados - GBD, mas não com o GBD específico.

II. Total dependência do GBD específico.

III. Não tem dependência com a classe do GBD a ser escolhido.

Definem os projetos de bancos de dados correta e respectivamente:

a) lógico, físico e conceitual.

b) lógico, conceitual e físico.

c) conceitual, lógico e físico.

d) físico, conceitual e lógico.

e) conceitual, físico e lógico

**Comentário:** Um **modelo conceitual** é uma descrição do banco de dados de forma independente de implementação em um SGBD. O modelo conceitual registra que dados podem aparecer no banco de dados, mas não registra como estes dados estão armazenados a nível de SGBD.

Já o **modelo lógico** é uma descrição de um banco de dados no nível de abstração visto pelo **usuário do SGBD**. Assim, o modelo lógico é dependente do tipo particular de SGBD que está sendo usado. Desta forma, temos uma restrição quanto ao tipo de SGBD que podemos utilizar.

**Detalhes de armazenamento interno** de informações, que não tem influência sobre a programação de aplicações no SGBD, mas podem influenciar no desempenho da aplicação (por exemplo, as estruturas de arquivos usadas no acesso às informações) não fazem parte do modelo lógico. Estes detalhes são representados no **modelo físico.** Neste caso, a restrição pode ser válida apenas para um SGBD específico.

Desta forma, podemos fazer o *match* entre as afirmações e as alternativas e encontrarmos nossa resposta na alternativa A.

**Gabarito: A**




**20. BANCA: FCC ANO: 2017 ÓRGÃO: TST PROVA: TÉCNICO JUDICIÁRIO – PROGRAMAÇÃO**

[56] Em um caso hipotético, um Programador do Tribunal Superior do Trabalho verificou que:

I. os dados nome do cidadão e número do processo não eram compartilhados entre três diferentes sistemas que os utilizavam;

II. não havia um sistema de log para acompanhamento e controle dos acessos aos bancos de dados de sua organização.

Os Sistemas Gerenciadores de Bancos de Dados – SGBDs contribuem para evitar que as situações referidas em I e II venham a ferir princípios que correspondem, respectivamente,

a) à replicação e à privacidade.

b) à redundância e à segurança lógica.

c) à integridade e à segurança física.

d) ao compartilhamento e à privacidade.

e) à concorrência e à integridade.

**Comentário:** O uso de banco de dados ajuda a **reduzir a redundância**. Em sistemas sem banco de dados, cada aplicação tem seus próprios arquivos privados. Esse fato pode levar a uma considerável redundância nos dados armazenados, com o resultante desperdício no espaço de armazenamento.

Outro ponto importante é a segurança dos dados. Neste sentido existe um controle de acesso, que visa autenticar e autorizar o acesso a base de dados. Contudo, é possível que um invasor consiga romper esse controle e ter acesso a sua base. Ou ate mesmo que ele seja um usuário cadastrado regularmente na sua estrutura de banco de dados. Nestes casos podemos optar por usar **uma segurança lógica**, que pode ser implementada por meio de uma trilha de auditoria. Trata-se de um arquivo de banco de dados especial em que o sistema automaticamente acompanha todas as operações realizadas por usuários sobre os dados. Em alguns sistemas, a trilha é integrada fisicamente ao log de recuperação.

Sendo assim, nossa resposta encontra-se na alternativa B.

**Gabarito: B**



**21. BANCA: FCC ANO: 2014 ÓRGÃO: TJ-AP PROVA: ANALISTA JUDICIÁRIO - BANCO DE DADOS - DBA**

Uma estrutura importante de um banco de dados relacional é o dicionário de dados, que

A não contempla o nome dos atributos de cada tabela.

B ocupa uma área máxima prevista de armazenamento de 1Kb.

C contém apenas a relação de usuários do banco de dados.

D deve conter o nome das tabelas presentes no banco de dados.

E é uma tabela que contém o histórico de utilização do banco de dados.

**Comentário:** Sem entrar nos detalhes das definições de um banco de dados relacional, apenas com o conhecimento que vimos até o momento, já temos a capacidade de responder à questão. Um dicionário de dados nos apresenta as informações ou descrições dos objetos presentes em um banco de dados. É preciso conhecer essas estruturas para pode manipulá-las.

Analisando as alternativas percebemos que nossa resposta, está na **alternativa D**, que é condizente com os conceitos básicos que apresentamos até aqui. Neste momento, não vou detalhar muito os erros presentes nas demais alternativas. Vamos, porém, tentar de forma rápida comentar as demais alternativas:

O dicionário de dados apresenta sim a descrição dos nomes das colunas ou atributos da tabela, por isso que a alternativa A está errada. A limitação do espaço para o dicionário de dados pode ser definida, varia de um SGBD para outro, mas, definitivamente, é bem superior a 1kb. O nome dos usuários está armazenado em tabelas do SGBD, contudo, não são as únicas informações armazenadas. Percebam ainda que essa informação não faz parte do dicionário de dados, ela é armazenada em uma tabela interna ou no banco de dados do próprio sistema. Por fim, a tabela de histórico ou auditoria são armazenadas de forma similar a uma tabela comum, onde os dados são separados dos metadados. A diferença é que nas tabelas de auditorias o SGBD ou outro aplicativo fica responsável por gravar os dados na tabela de histórico quando uma alteração é feita na tabela monitorada.

Eu disse que seria rápido e acabei tomando algumas linhas, se você não assimilou muito bem o que está escrito no parágrafo anterior, não se preocupe, siga em frente!

**Gabarito: D.**

---

CAIU na prova!

**22. BANCA: FCC ANO: 2014 ÓRGÃO: TCE-GO PROVA: ANALISTA DE CONTROLE EXTERNO - TECNOLOGIA DA INFORMAÇÃO**

Ao utilizar o Banco de Dados torna-se necessário controlar os dados, sua manipulação e os seus usuários. Isso é feito através de um conjunto de aplicações de software que

a) permitem o acesso aos dados de maneira simplificada;

b) autorizam os acessos de múltiplos usuários às informações;

c) controlam a manipulação dos dados presentes no banco de dados (inserção, supressão, modificação etc.).

Esses serviços são providos por

A um ERP.

B técnicas de Data Mining.

C um SGBD.

D um administrador de Banco de Dados.

E um sistema de dados multidimensional.

**Comentário:** Veja que estamos tratando de um conjunto de aplicações de software. Vimos que o SGBD é composto por um conjunto de software. Entre eles temos alguns sistemas relacionados com a segurança do banco de dados. Uma das tarefas de segurança é garantir o acesso ao banco de dados, para isso é necessário um usuário autorizado. Depois é preciso definir os privilégios para acesso e manipulação dos objetos. Este privilégio pode ser garantido a um usuário diretamente aos objetos ou por meio de ROLES. O SGBD possui funções ou programas responsáveis por estruturar as opções de acesso e de permissões aos diferentes usuários do banco de dados.

Para não perdemos o hábito vamos comentar sobre os conceitos que aparecem nas alternativas e ainda não foram vistos no nosso curso.

**ERP** é uma sigla derivada do nome *Enterprise Resource Planning*. ERPs são que softwares que integram todos os dados e processos de uma organização em um único sistema. Em termos gerais, um sistema de ERP é uma plataforma de software desenvolvida para integrar os diversos departamentos de uma empresa possibilitando a automação e armazenamento de todas as informações de negócios. O ERP possibilita um fluxo de informações único, contínuo e consistente.

Mineração de dados, ou data mining, é o processo de análise de conjuntos de dados que tem por objetivo a descoberta de padrões interessantes e que possam representar informações úteis. A **técnica de mineração** consiste na especificação de métodos que nos garanta como descobrir os padrões que nos interessam. Dentre as principais técnicas utilizadas em mineração de dados, temos: técnicas estatísticas, técnicas de aprendizado de máquina e técnicas baseadas em crescimento-poda-validação.

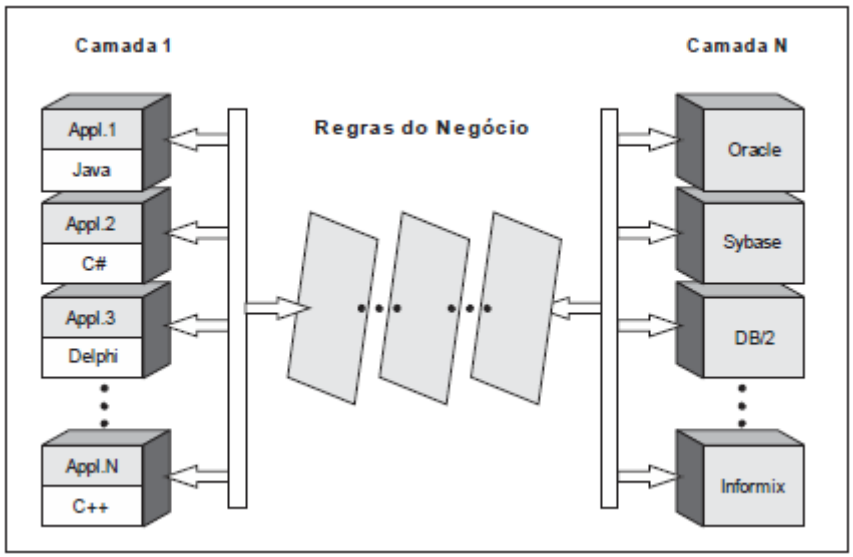
Um **modelo dimensional** contém as mesmas informações que um modelo normalizado. Os pacotes para dados multidimensionais oferecem um formato com as seguintes preocupações: facilidade de compreensão ao usuário, desempenho da consulta e resiliência à mudança. Esse tipo de modelo é composto por tabelas fato e dimensões.

**Gabarito: C.**

CAIU na prova!

**23. BANCA: FCC ANO: 2015 ÓRGÃO: MANAUSPREV PROVA: ANALISTA PREVIDENCIÁRIO - TECNOLOGIA DA INFORMAÇÃO**

Considere a arquitetura de um SGBD mostrada na figura abaixo.




Trata-se de uma arquitetura

A distribuída em N camadas, em que a informação está distribuída em diversos servidores. Cada servidor atua como no sistema cliente-servidor, porém as consultas oriundas dos aplicativos são feitas para qualquer servidor indistintamente, através da rede.

B centralizada, em que existe um computador com grande capacidade de processamento, que é o hospedeiro do SGBD e emuladores para os vários aplicativos. Tem como principal vantagem o baixo custo, pois permite que muitos usuários manipulem grande volume de dados.

C cliente-servidor, em que o cliente executa as tarefas do aplicativo, ou seja, fornece a interface do usuário. O servidor executa as consultas no SGBD e retorna os resultados ao cliente, aumentando o tráfego da rede.

D descentralizada, pois o sistema encarrega-se de obter a informação necessária, de maneira transparente para o aplicativo, que passa a atuar consultando seu servidor. Porém, é dependente de aspectos lógicos de carga de acesso aos dados, o que a torna desvantajosa.

E em rede, em que a base de dados é fortemente acoplada. Sua vantagem é que cada aplicativo acessa apenas o servidor que dispõe dos seus dados, através do acesso à rede.

**Comentário.** Para respondermos a questão com segurança vamos apresentar abaixo um resumo das principais arquiteturas de SGBDs.

**Plataformas centralizadas:** Na arquitetura centralizada, existe um computador com grande capacidade de processamento, que é o hospedeiro do SGBD e emuladores para os vários aplicativos. Esta arquitetura tem como principal vantagem a de permitir que muitos usuários manipulem grande volume de dados. Sua principal desvantagem está no alto custo, pois exige ambiente especial para mainframes e soluções centralizadas.

**Sistemas de Computador Pessoal - PC:** Os computadores pessoais trabalham em sistemas stand-alone, ou seja, fazem seus processamentos sozinhos. No princípio esse processamento era bastante limitado, porém, com a evolução do hardware, hoje temos PCs com grande capacidade de processamento. Eles utilizam o padrão Xbase e quando se trata de SGBDs, funcionam como hospedeiros e terminais. Desta maneira, possuem um único aplicativo a ser executado na máquina. A principal vantagem desta arquitetura é a simplicidade.

**Banco de Dados Cliente-Servidor:** Na arquitetura Cliente-Servidor, o cliente (*front_end*) executa as tarefas do aplicativo, ou seja, fornece a interface do usuário (tela, e processamento de entrada e saída). O servidor (*back_end*) executa as consultas no DBMS e retorna os resultados ao cliente. Apesar de ser uma arquitetura bastante popular, são necessárias soluções sofisticadas de software que possibilitem: o tratamento de transações, as confirmações de transações (*commits*), desfazer transações (*rollbacks*), linguagens de consultas (*stored procedures*) e gatilhos (*triggers*). A principal vantagem desta arquitetura é a divisão do processamento entre dois sistemas, o que reduz o tráfego de dados na rede.

**Banco de Dados Distribuídos (N camadas):** Nesta arquitetura, a informação está distribuída em diversos servidores. Como exemplo, observe a abaixo. Cada servidor atua como no sistema cliente-servidor, porém as consultas oriundas dos aplicativos são feitas para qualquer servidor indistintamente. Caso a informação solicitada seja mantida por outro servidor ou servidores, o sistema encarrega-se de obter a informação necessária, de maneira transparente para o aplicativo, que passa a atuar consultando a rede, independente de conhecer seus servidores. Exemplos típicos são as bases de dados corporativas, em que o volume de informação é muito grande e, por isso, deve ser distribuído em diversos servidores. Porém, não é dependente de aspectos lógicos de carga de acesso aos dados, ou base de dados fracamente acopladas, em que uma informação solicitada vai sendo coletada numa propagação da consulta numa cadeia de servidores. A característica básica é a existência de diversos programas aplicativos consultando a rede para acessar os dados necessários, porém, sem o conhecimento explícito de quais servidores dispõem desses dados.

**Gabarito: A.**

---




**24. BANCA: FCC ANO: 2013 ÓRGÃO: MPE-SE PROVA: ANALISTA DO MINISTÉRIO PÚBLICO - GESTÃO E ANÁLISE DE PROJETO DE SISTEMA**

Em projetos de Banco de Dados, o objetivo da arquitetura de três-esquemas é separar o usuário da aplicação do banco de dados físico. Nessa arquitetura, os esquemas podem ser definidos por três níveis:

I. O nível interno tem um esquema que descreve a estrutura de armazenamento físico do banco de dados. Esse esquema utiliza um modelo de dado físico e descreve os detalhes complexos do armazenamento de dados e caminhos de acesso ao banco;

II. O nível conceitual possui um esquema que descreve a estrutura de todo o banco de dados para a comunidade de usuários. O esquema conceitual oculta os detalhes das estruturas de armazenamento físico e se concentra na descrição de entidades, tipos de dados, conexões, operações de usuários e restrições. Geralmente, um modelo de dados representacional é usado para descrever o esquema conceitual quando o sistema de banco de dados for implementado. Esse esquema de implementação conceitual é normalmente baseado em um projeto de esquema conceitual em um modelo de dados de alto nível;

III. O nível interno ainda abrange os esquemas externos ou visões de usuários. Cada esquema interno descreve a parte do banco de dados que um dado grupo de usuários tem interesse e oculta o restante do banco de dados desse grupo. Como no item anterior, cada esquema é tipicamente implementado usando-se um modelo de dados representacional, possivelmente baseado em um projeto de esquema externo em um modelo de dados de alto nível.

Está correto o que se afirma em

A II, apenas.

B II e III, apenas.

C I, II e III.

D I e II, apenas.

E III, apenas.

**Comentário:** Vamos analisar cada uma das alternativas acima.

Na alternativa I temos uma definição de um esquema físico. Percebam que um esquema é descrito conforme um modelo de dados. Neste caso, estamos falando de um modelo de dados físico ou de baixo nível. A alternativa está **correta**.

A alternativa II apresenta o modelo conceitual da arquitetura em três esquemas. Lembre-se que neste caso ele é visto com a categoria do meio na estrutura hierárquica. O texto está perfeito e a alternativa encontra-se **correta**.

A alternativa III, no entanto, encontra-se **errada**. O texto tenta confundir o candidato descrevendo o nível externo ou de visão e associando a definição ao nível interno. Já falamos sobre nível interno na alternativa I.

**Gabarito: D.**




**25. BANCA: FCC ANO: 2014 ÓRGÃO: TJ-AP PROVA: ANALISTA JUDICIÁRIO - BANCO DE DADOS - DBA**

A adoção de regras de nomenclatura dos dados de um banco de dados é um tema que vem sendo considerado por muitas empresas. Caso uma empresa opte por normatizar e padronizar os dados em seus bancos de dados, é correto afirmar que

A as normas, por serem rígidas, podem levar à duplicação de informações.

B embora traga algumas vantagens, essa prática leva à proliferação de dados incompatíveis e redundantes.

C facilitará a manutenção e a organização dos códigos e diretórios.

D implicará em custos elevados, pois o mecanismo de aplicação das normas tende a duplicar o tempo de desenvolvimento.

E o acesso ao banco de dados ficará mais lento, pois um servidor de normatização deverá ser implantado.

**Comentário:** A questão de normatização das informações é uma tarefa relevante e complexa na vida do administrador de dados. Primeiramente, é necessário definir um conjunto de regras sintáticas que devem ser seguidas para que o modelo possa ser avaliado automaticamente pelas ferramentas de modelagem.

Outra tarefa é a descrição de um repositório central de dados, conhecido como dados mestres da organização. Esses dados são disponibilizados para todos os usuários que necessitarem. A ideia de normatizar essas informações genéricas deve reduzir a duplicação de informações e facilitar a manutenção dos dados, mantendo a qualidade dos dados.

Esses dois aspectos do controle e normatização dos dados nos leva a uma melhor organização dos códigos e diretórios. Desta forma, podemos verificar nossa resposta na alternativa C.

Sobre a implementação desses critérios de avaliação eles ficaram dispersos na infraestrutura. A parte sintática, regras e nomenclatura, pode ser avaliada pela ferramenta de modelagem (e.g.: *Power Designer*). O nível de dispersão e a qualidade dos dados pode ser concretizado por uma ferramenta de *Data Quality*.

**Gabarito: C.**

---




**26. BANCA: FCC ANO: 2012 ÓRGÃO: TRE-SP PROVA: TÉCNICO DO JUDICIÁRIO - PROGRAMADOR DE SISTEMAS**

Em SGBDs,

A os metadados resultam da conversão de comandos DDL pelo compilador da DDL.

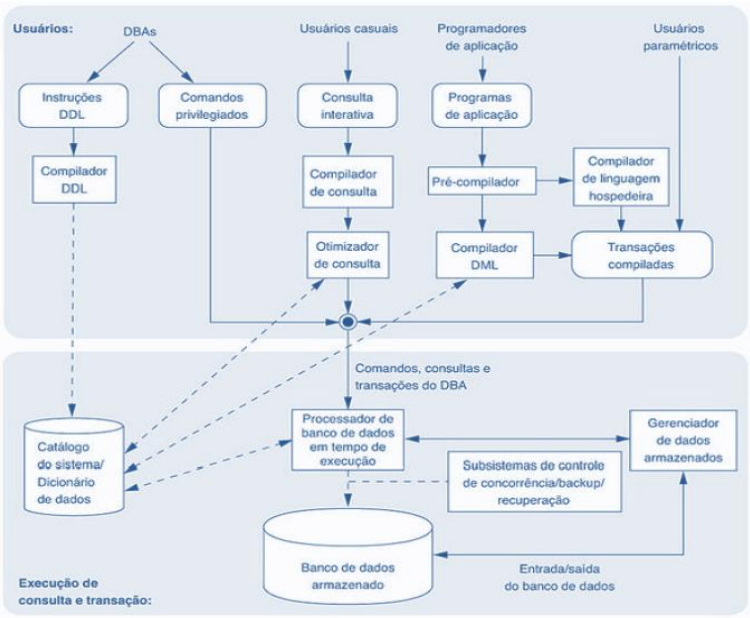
B mapeamentos, restrições de integridade, mensagens de comunicação e restrições de segurança são recursos contidos no dicionário de dados.

C o arquivo de dados é o componente que cuida da alocação do espaço na armazenagem no disco e das estruturas de dados usadas para representar a informação armazenada.

D a estrutura de armazenagem e os métodos de acesso são especificados por um conjunto de definições em um tipo especial de DML.

E a abstração dos níveis físico, conceitual e de visão aplica-se, exclusivamente, à definição e estrutura de dados.

**Comentário:** Para responder essa questão vamos recorrer a ajuda da figura a seguir, retira do livro do Navathe:




Percebam que os DBAs definem o banco de dados e realizam ajustes, alterando sua definição por meio da DDL e de outros comandos privilegiados. O compilador da DDL processa as definições de esquema especificadas e armazena as descrições dos esquemas (metadados) no catálogo do SGBD. Vejam que essa descrição está em acordo com a alternativa A, que, portanto, é a nossa resposta. Vejamos agora os erros das demais alternativas.

A alternativa B trata de forma incorreta a função do catálogo de dados. O catálogo, na realidade, inclui informações como os nomes e os tamanhos dos arquivos, nomes e topos de dados dos itens de ados, detalhes de armazenamento de cada arquivo, informações de mapeamento entre os esquemas e restrições.

A alternativa C tenta atribuir ao arquivo de dados uma função que não é dele. O arquivo de dados, basicamente, armazena os registros cujas estruturas são definidas no catálogo de dados.

Sabemos que DML está relacionado à manipulação de dados e não à definição de estruturas, essa seria descritas por um conjunto de comandos denominados DDL (*data definition language*). Eis o erro da alternativa D.

Para responder a alternativa E podemos recorrer ao livro do Navathe: "o SGBD precisa transformar uma solicitação especificada em um esquema externo em uma solicitação no esquema conceitual, e depois em uma solicitação no esquema interno para processamento no banco de dados armazenado. Se a solicitação for uma recuperação, os dados extraídos do banco de dados armazenado devem ser reformatados para corresponder à visão externa dos usuários". Percebam que a utilização de hierarquia também tem efeitos sobre a manipulação dos dados.

**Gabarito: A.**

---




**27. BANCA: FCC ANO: 2013 ÓRGÃO: MPE-SE PROVA: ANALISTA DO MINISTÉRIO PÚBLICO - GESTÃO E ANÁLISE DE PROJETO DE SISTEMA**

A capacidade de alterar o esquema conceitual sem mudar o esquema externo ou os programas, podendo modificar o esquema conceitual para expandir o banco de dados (adicionando um tipo de registro ou item de dados), variar as restrições ou reduzir o banco de dados (removendo um tipo de registro ou item de dados) é chamada de

A modularidade.

B modelo conceitual.

C independência lógica de dados.

D polimorfismo.

E agregação.

**Comentário:** Vamos aproveitar a questão para relembra da definição de independência física e lógica de dados.

**Independência lógica de dados** é a capacidade de alterar o esquema conceitual sem ter de alterar os esquemas externos ou os programas de aplicação. Podemos alterar o esquema conceitual para expandir o banco de dados, para alterar restrições ou para reduzir o banco de dados. Somente a definição da visão e os mapeamentos precisam ser alterados em um SGBD que suporta a independência lógica de dados.

**Independência física de dados** é a capacidade de alterar o esquema interno sem ter de alterar o esquema conceitual. Logo, os esquemas externos também não precisam ser alterados. Mudanças no esquema interno podem ser necessárias porque alguns arquivos físicos foram reorganizados para melhorar o desempenho da recuperação ou atualização.

Analisando as definições acima, podemos marcar nosso gabarito na alternativa C.

**Gabarito: C.**




**28. BANCA: FCC ANO: 2012 ÓRGÃO: TCE-AM PROVA: ANALISTA TÉCNICO DE CONTROLE EXTERNO - TECNOLOGIA DA INFORMAÇÃO**

O modelo conceitual de dados

A é aquele em que os objetos, suas características e relacionamentos têm a representação de acordo com as regras de implementação e limitantes impostos por algum tipo de tecnologia.

B é aquele em que os objetos, suas características e relacionamentos têm a representação fiel ao ambiente observado, independente de limitações quaisquer impostas por tecnologias, técnicas de implementação ou dispositivos físicos.

C é aquele elaborado respeitando-se e implementando-se conceitos tais como chaves de acesso, controle de chaves duplicadas, itens de repetição (arrays), normalização, ponteiros e integridade referencial, entre outros.

D é a fase da modelagem na qual é necessário considerar todas as particularidades de implementação, principalmente o modelo que será utilizado para a implementação futura.

E está sempre associado às fases de projeto, contrastando com o modelo lógico, que sempre está associado à fase de análise, quando utilizado com as metodologias de desenvolvimento de sistemas e implementado por ferramentas CASE.

**Comentário:** A modelagem conceitual é uma fase muito importante no projeto de uma aplicação de banco de dados bem-sucedida. A criação de um esquema conceitual deve utilizar um modelo de dados conceitual ou de alto nível.

O esquema conceitual é uma descrição concisa dos requisitos de dados dos usuários e inclui detalhes dos tipos de entidades, relacionamentos e restrições; estes são expressos usando os conceitos fornecidos pelo modelo de dados.

Essa técnica permite que os projetistas de banco de dados se concentrem em especificar as propriedades dos dados, sem se preocupar com detalhes de armazenamento e implementação. Em outras palavras, nesta etapa os artefatos gerados são independentes de SGBD.

Usando o contexto teórico descrito nas linhas acima, podemos encontrar nossa resposta na alternativa B.

**Gabarito: B.**

CAIU na prova!

**29. BANCA: FCC ANO: 2009 ÓRGÃO: TRT - 3ª REGIÃO (MG) PROVA: ANALISTA JUDICIÁRIO - TECNOLOGIA DA INFORMAÇÃO**

De acordo com a Regra Áurea, nenhuma operação de atualização poderá atribuir a qualquer banco de dados um valor que faça com que seu predicado (restrição declarada formalmente) de banco de dados seja

A avaliado como TRUE.

B avaliado como FALSE.

C avaliado como DOUBLE.

D maior que 1.

E menor que 1.

**Comentário:** A definição de **regra de ouro** ou **Regra Áurea** é que nenhuma operação de atualização deve atribuir para qualquer banco de dados um valor que faça uma das suas restrições ou ***constraint*** ser **avaliada como falsa**. Em outras palavras, não deve ser possível mudar o valor de um atributo de uma tabela atualizando a informação com um novo valor que fere as restrições definidas no momento da criação da tabela. Essas restrições também podem ser alteradas por meio do comando ALTER TABLE. O fato é que a integridade e consistência de uma tabela depende do respeito as restrições associadas a elas. Desta forma, podemos marcar nosso gabarito na alternativa B.

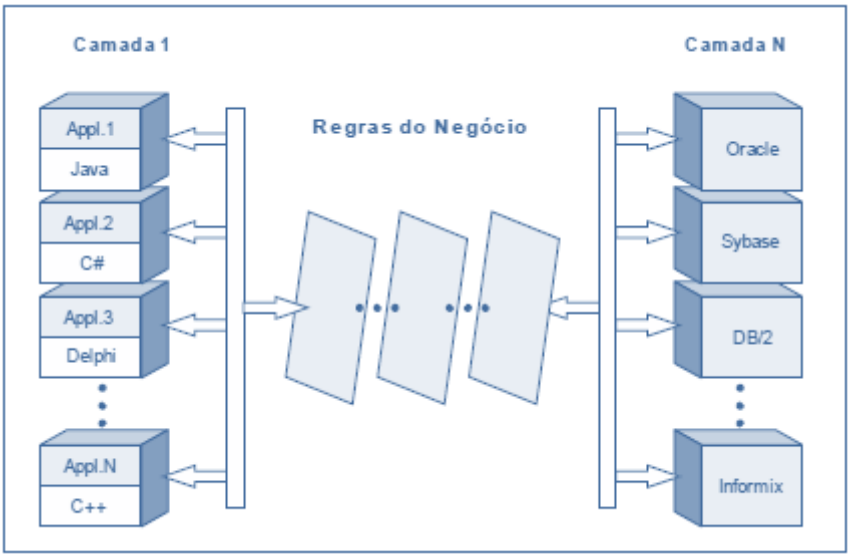
**Gabarito: B.**

---

CAIU na prova!

**30. BANCA: FCC ANO: 2015 ÓRGÃO: MANAUSPREV PROVA: ANALISTA PREVIDENCIÁRIO - TECNOLOGIA DA INFORMAÇÃO**

Considere a arquitetura de um SGBD mostrada na figura abaixo.




Trata-se de uma arquitetura

A distribuída em N camadas, em que a informação está distribuída em diversos servidores. Cada servidor atua como no sistema cliente-servidor, porém as consultas oriundas dos aplicativos são feitas para qualquer servidor indistintamente, através da rede.

B centralizada, em que existe um computador com grande capacidade de processamento, que é o hospedeiro do SGBD e emuladores para os vários aplicativos. Tem como principal vantagem o baixo custo, pois permite que muitos usuários manipulem grande volume de dados.

C cliente-servidor, em que o cliente executa as tarefas do aplicativo, ou seja, fornece a interface do usuário. O servidor executa as consultas no SGBD e retorna os resultados ao cliente, aumentando o tráfego da rede.

D descentralizada, pois o sistema encarrega-se de obter a informação necessária, de maneira transparente para o aplicativo, que passa a atuar consultando seu servidor. Porém, é dependente de aspectos lógicos de carga de acesso aos dados, o que a torna desvantajosa.

E em rede, em que a base de dados é fortemente acoplada. Sua vantagem é que cada aplicativo acessa apenas o servidor que dispõe dos seus dados, através do acesso à rede.

**Comentário.** Para respondermos a questão com segurança vamos apresentar abaixo um resumo das principais arquiteturas de SGBDs.

**Plataformas centralizadas:** Na arquitetura centralizada, existe um computador com grande capacidade de processamento, que é o hospedeiro do SGBD e emuladores para os vários aplicativos. Esta arquitetura tem como principal vantagem a de permitir que muitos usuários manipulem grande volume de dados. Sua principal desvantagem está no alto custo, pois exige ambiente especial para mainframes e soluções centralizadas.

**Sistemas de Computador Pessoal - PC:** Os computadores pessoais trabalham em sistemas stand-alone, ou seja, fazem seus processamentos sozinhos. No princípio esse processamento era bastante limitado, porém, com a evolução do hardware, hoje temos PCs com grande capacidade de processamento. Eles utilizam o padrão Xbase e quando se trata de SGBDs,

funcionam como hospedeiros e terminais. Desta maneira, possuem um único aplicativo a ser executado na máquina. A principal vantagem desta arquitetura é a simplicidade.

**Banco de Dados Cliente-Servidor:** Na arquitetura Cliente-Servidor, o cliente (*front_end*) executa as tarefas do aplicativo, ou seja, fornece a interface do usuário (tela, e processamento de entrada e saída). O servidor (*back_end*) executa as consultas no DBMS e retorna os resultados ao cliente. Apesar de ser uma arquitetura bastante popular, são necessárias soluções sofisticadas de software que possibilitem: o tratamento de transações, as confirmações de transações (*commits*), desfazer transações (*rollbacks*), linguagens de consultas (*stored procedures*) e gatilhos (*triggers*). A principal vantagem desta arquitetura é a divisão do processamento entre dois sistemas, o que reduz o tráfego de dados na rede.

**Banco de Dados Distribuídos (N camadas):** Nesta arquitetura, a informação está distribuída em diversos servidores. Como exemplo, observe a abaixo. Cada servidor atua como no sistema cliente-servidor, porém as consultas oriundas dos aplicativos são feitas para qualquer servidor indistintamente. Caso a informação solicitada seja mantida por outro servidor ou servidores, o sistema encarrega-se de obter a informação necessária, de maneira transparente para o aplicativo, que passa a atuar consultando a rede, independente de conhecer seus servidores. Exemplos típicos são as bases de dados corporativas, em que o volume de informação é muito grande e, por isso, deve ser distribuído em diversos servidores. Porém, não é dependente de aspectos lógicos de carga de acesso aos dados, ou base de dados fracamente acopladas, em que uma informação solicitada vai sendo coletada numa propagação da consulta numa cadeia de servidores. A característica básica é a existência de diversos programas aplicativos consultando a rede para acessar os dados necessários, porém, sem o conhecimento explícito de quais servidores dispõem desses dados.

**Gabarito: A.**

## EXERCÍCIOS




**1. Ano: 2018 Banca: FCC Órgão: DPE-AM Cargo: Analista Área: Banco de Dados Questão: 47**

Uma das propriedades conhecidas em bancos de dados relacionais como ACID é a durabilidade, segundo a qual, em uma transação completada com sucesso, todas as atualizações feitas no banco de dados por essa transação devem persistir

(A) com exceção da ocorrência de falhas na memória do servidor de banco de dados.

(B) mesmo na ocorrência de falhas no sistema de banco de dados, após o término da transação.

(C) a menos que ocorra uma falha no servidor em até 1 segundo, após o término da transação.

(D) exceto se houver outras transações em execução no momento da falha.

(E) com exceção da ocorrência de falhas no barramento de dados do servidor do banco de dados.




**2. Ano: 2018 Banca: FCC Órgão: DPE-AM Cargo: Analista Área: Banco de Dados Questão: 24**

Dentre os diversos tipos de bancos de dados, existe o denominado bancos de dados objeto-relacionais, que tem como fundamento a

(A) duplicação das tabelas componentes de um banco de dados originalmente relacional.

(B) incorporação de características e recursos da orientação a objetos nos bancos de dados originalmente relacionais.

(C) eliminação do conceito de atributos, existente nos bancos de dados originalmente relacionais.

(D) transformação de todas as tabelas de um banco de dados originalmente relacional em uma única classe da orientação a objetos.

(E) substituição do conceito representado pelas propriedades ACID pelo teorema CAP.



**3. BANCA: FCC ANO: 2017 ÓRGÃO: TST PROVA: TÉCNICO JUDICIÁRIO – PROGRAMAÇÃO**

[57] Considere que um determinado tipo de banco de dados organiza os dados na forma de uma pirâmide, onde o registro principal no topo da pirâmide é chamado registro raiz. Os registros são organizados como pai e filho onde um registro filho sempre tem apenas um registro pai ao qual ele está ligado, como em uma árvore familiar normal. Em contrapartida, um registro pai pode ter mais de um registro filho a ele ligado.

Trata-se do Banco de Dados

a) hierárquico.

b) relacional.

c) orientado a objeto.

d) objeto-relacional.

e) de rede.




**4. BANCA: FCC ANO: 2017 ÓRGÃO: DPE-RS PROVA: ANALISTA – BANCO DE DADOS**

[37] O dicionário de dados de um banco de dados relacional

a) não se aplica a tabelas com pequeno número de registros.

b) não considera o armazenamento da lista de atributos chave das tabelas.

c) armazena, dentre outras informações, nomes de tabelas e de seus atributos.

d) ignora os domínios de cada atributo das tabelas.

e) somente se aplica a tabelas com grande número de atributos.




**5. BANCA: FCC ANO: 2016 ÓRGÃO: PREFEITURA DE TERESINA - PI PROVA: TÉCNICO DE NÍVEL SUPERIOR - ANALISTA DE SISTEMAS**

[58] Em relação a projetos de bancos de dados, considere:

I. Tem dependência com a classe do Gerenciador de Banco de Dados - GBD, mas não com o GBD específico.

II. Total dependência do GBD específico.

III. Não tem dependência com a classe do GBD a ser escolhido.

Definem os projetos de bancos de dados correta e respectivamente:

a) lógico, físico e conceitual.

b) lógico, conceitual e físico.

c) conceitual, lógico e físico.

d) físico, conceitual e lógico.

e) conceitual, físico e lógico




**6. BANCA: FCC ANO: 2017 ÓRGÃO: TST PROVA: TÉCNICO JUDICIÁRIO – PROGRAMAÇÃO**

[56] Em um caso hipotético, um Programador do Tribunal Superior do Trabalho verificou que:

I. os dados nome do cidadão e número do processo não eram compartilhados entre três diferentes sistemas que os utilizavam;

II. não havia um sistema de log para acompanhamento e controle dos acessos aos bancos de dados de sua organização.

Os Sistemas Gerenciadores de Bancos de Dados – SGBDs contribuem para evitar que as situações referidas em I e II venham a ferir princípios que correspondem, respectivamente,

a) à replicação e à privacidade.

b) à redundância e à segurança lógica.

c) à integridade e à segurança física.

d) ao compartilhamento e à privacidade.

e) à concorrência e à integridade.




**7. BANCA: FCC ANO: 2014 ÓRGÃO: TJ-AP PROVA: ANALISTA JUDICIÁRIO - BANCO DE DADOS - DBA**

Uma estrutura importante de um banco de dados relacional é o dicionário de dados, que

A não contempla o nome dos atributos de cada tabela.

B ocupa uma área máxima prevista de armazenamento de 1Kb.

C contém apenas a relação de usuários do banco de dados.

D deve conter o nome das tabelas presentes no banco de dados.

E é uma tabela que contém o histórico de utilização do banco de dados.




**8. BANCA: FCC ANO: 2014 ÓRGÃO: TCE-GO PROVA: ANALISTA DE CONTROLE EXTERNO - TECNOLOGIA DA INFORMAÇÃO**

Ao utilizar o Banco de Dados torna-se necessário controlar os dados, sua manipulação e os seus usuários. Isso é feito através de um conjunto de aplicações de software que

a) permitem o acesso aos dados de maneira simplificada;

b) autorizam os acessos de múltiplos usuários às informações;

c) controlam a manipulação dos dados presentes no banco de dados (inserção, supressão, modificação etc.).

Esses serviços são providos por

A um ERP.

B técnicas de Data Mining.

C um SGBD.

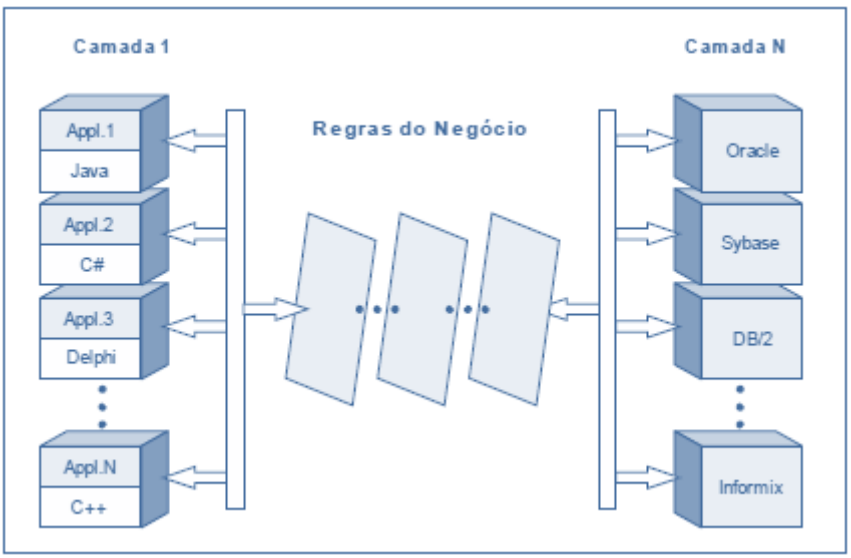
D um administrador de Banco de Dados.

E um sistema de dados multidimensional.

CAIU na prova!

**9. BANCA: FCC ANO: 2015 ÓRGÃO: MANAUSPREV PROVA: ANALISTA PREVIDENCIÁRIO - TECNOLOGIA DA INFORMAÇÃO**

Considere a arquitetura de um SGBD mostrada na figura abaixo.




Trata-se de uma arquitetura

A distribuída em N camadas, em que a informação está distribuída em diversos servidores. Cada servidor atua como no sistema cliente-servidor, porém as consultas oriundas dos aplicativos são feitas para qualquer servidor indistintamente, através da rede.

B centralizada, em que existe um computador com grande capacidade de processamento, que é o hospedeiro do SGBD e emuladores para os vários aplicativos. Tem como principal vantagem o baixo custo, pois permite que muitos usuários manipulem grande volume de dados.

C cliente-servidor, em que o cliente executa as tarefas do aplicativo, ou seja, fornece a interface do usuário. O servidor executa as consultas no SGBD e retorna os resultados ao cliente, aumentando o tráfego da rede.

D descentralizada, pois o sistema encarrega-se de obter a informação necessária, de maneira transparente para o aplicativo, que passa a atuar consultando seu servidor. Porém, é dependente de aspectos lógicos de carga de acesso aos dados, o que a torna desvantajosa.

E em rede, em que a base de dados é fortemente acoplada. Sua vantagem é que cada aplicativo acessa apenas o servidor que dispõe dos seus dados, através do acesso à rede.




**10. BANCA: FCC ANO: 2013 ÓRGÃO: MPE-SE PROVA: ANALISTA DO MINISTÉRIO PÚBLICO - GESTÃO E ANÁLISE DE PROJETO DE SISTEMA**

Em projetos de Banco de Dados, o objetivo da arquitetura de três-esquemas é separar o usuário da aplicação do banco de dados físico. Nessa arquitetura, os esquemas podem ser definidos por três níveis:

I. O nível interno tem um esquema que descreve a estrutura de armazenamento físico do banco de dados. Esse esquema utiliza um modelo de dado físico e descreve os detalhes complexos do armazenamento de dados e caminhos de acesso ao banco;

II. O nível conceitual possui um esquema que descreve a estrutura de todo o banco de dados para a comunidade de usuários. O esquema conceitual oculta os detalhes das estruturas de armazenamento físico e se concentra na descrição de entidades, tipos de dados, conexões, operações de usuários e restrições. Geralmente, um modelo de dados representacional é usado para descrever o esquema conceitual quando o sistema de banco de dados for implementado. Esse esquema de implementação conceitual é normalmente baseado em um projeto de esquema conceitual em um modelo de dados de alto nível;

III. O nível interno ainda abrange os esquemas externos ou visões de usuários. Cada esquema interno descreve a parte do banco de dados que um dado grupo de usuários tem interesse e oculta o restante do banco de dados desse grupo. Como no item anterior, cada esquema é tipicamente implementado usando-se um modelo de dados representacional, possivelmente baseado em um projeto de esquema externo em um modelo de dados de alto nível.

Está correto o que se afirma em

A II, apenas.

B II e III, apenas.

C I, II e III.

D I e II, apenas.

E III, apenas.




**11. BANCA: FCC ANO: 2014 ÓRGÃO: TJ-AP PROVA: ANALISTA JUDICIÁRIO - BANCO DE DADOS - DBA**

A adoção de regras de nomenclatura dos dados de um banco de dados é um tema que vem sendo considerado por muitas empresas. Caso uma empresa opte por normatizar e padronizar os dados em seus bancos de dados, é correto afirmar que

A as normas, por serem rígidas, podem levar à duplicação de informações.

B embora traga algumas vantagens, essa prática leva à proliferação de dados incompatíveis e redundantes.

C facilitará a manutenção e a organização dos códigos e diretórios.

D implicará em custos elevados, pois o mecanismo de aplicação das normas tende a duplicar o tempo de desenvolvimento.

E o acesso ao banco de dados ficará mais lento, pois um servidor de normatização deverá ser implantado.




**12. BANCA: FCC ANO: 2012 ÓRGÃO: TRE-SP PROVA: TÉCNICO DO JUDICIÁRIO - PROGRAMADOR DE SISTEMAS**

Em SGBDs,

A os metadados resultam da conversão de comandos DDL pelo compilador da DDL.

B mapeamentos, restrições de integridade, mensagens de comunicação e restrições de segurança são recursos contidos no dicionário de dados.

C o arquivo de dados é o componente que cuida da alocação do espaço na armazenagem no disco e das estruturas de dados usadas para representar a informação armazenada.

D a estrutura de armazenagem e os métodos de acesso são especificados por um conjunto de definições em um tipo especial de DML.

E a abstração dos níveis físico, conceitual e de visão aplica-se, exclusivamente, à definição e estrutura de dados.



**13. BANCA: FCC ANO: 2013 ÓRGÃO: MPE-SE PROVA: ANALISTA DO MINISTÉRIO PÚBLICO - GESTÃO E ANÁLISE DE PROJETO DE SISTEMA**

A capacidade de alterar o esquema conceitual sem mudar o esquema externo ou os programas, podendo modificar o esquema conceitual para expandir o banco de dados (adicionando um tipo de registro ou item de dados), variar as restrições ou reduzir o banco de dados (removendo um tipo de registro ou item de dados) é chamada de

A modularidade.

B modelo conceitual.

C independência lógica de dados.

D polimorfismo.

E agregação.




**14. BANCA: FCC ANO: 2012 ÓRGÃO: TCE-AM PROVA: ANALISTA TÉCNICO DE CONTROLE EXTERNO - TECNOLOGIA DA INFORMAÇÃO**

O modelo conceitual de dados

A é aquele em que os objetos, suas características e relacionamentos têm a representação de acordo com as regras de implementação e limitantes impostos por algum tipo de tecnologia.

B é aquele em que os objetos, suas características e relacionamentos têm a representação fiel ao ambiente observado, independente de limitações quaisquer impostas por tecnologias, técnicas de implementação ou dispositivos físicos.

C é aquele elaborado respeitando-se e implementando-se conceitos tais como chaves de acesso, controle de chaves duplicadas, itens de repetição (arrays), normalização, ponteiros e integridade referencial, entre outros.

D é a fase da modelagem na qual é necessário considerar todas as particularidades de implementação, principalmente o modelo que será utilizado para a implementação futura.

E está sempre associado às fases de projeto, contrastando com o modelo lógico, que sempre está associado à fase de análise, quando utilizado com as metodologias de desenvolvimento de sistemas e implementado por ferramentas CASE.




**15. BANCA: FCC ANO: 2009 ÓRGÃO: TRT - 3ª REGIÃO (MG) PROVA: ANALISTA JUDICIÁRIO - TECNOLOGIA DA INFORMAÇÃO**

De acordo com a Regra Áurea, nenhuma operação de atualização poderá atribuir a qualquer banco de dados um valor que faça com que seu predicado (restrição declarada formalmente) de banco de dados seja

A avaliado como TRUE.

B avaliado como FALSE.

C avaliado como DOUBLE.

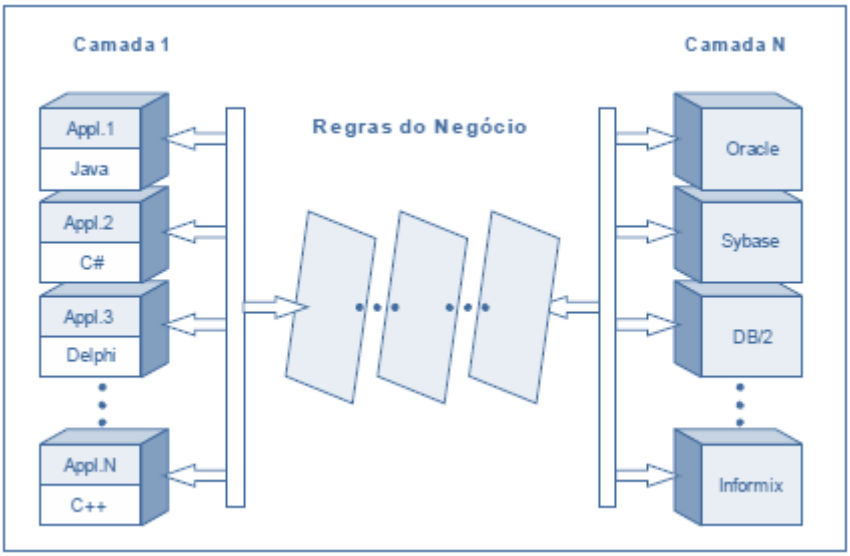
D maior que 1.

E menor que 1.

CAIU na prova!

**16. BANCA: FCC ANO: 2015 ÓRGÃO: MANAUSPREV PROVA: ANALISTA PREVIDENCIÁRIO - TECNOLOGIA DA INFORMAÇÃO**

Considere a arquitetura de um SGBD mostrada na figura abaixo.




Trata-se de uma arquitetura

A distribuída em N camadas, em que a informação está distribuída em diversos servidores. Cada servidor atua como no sistema cliente-servidor, porém as consultas oriundas dos aplicativos são feitas para qualquer servidor indistintamente, através da rede.

B centralizada, em que existe um computador com grande capacidade de processamento, que é o hospedeiro do SGBD e emuladores para os vários aplicativos. Tem como principal vantagem o baixo custo, pois permite que muitos usuários manipulem grande volume de dados.

C cliente-servidor, em que o cliente executa as tarefas do aplicativo, ou seja, fornece a interface do usuário. O servidor executa as consultas no SGBD e retorna os resultados ao cliente, aumentando o tráfego da rede.

D descentralizada, pois o sistema encarrega-se de obter a informação necessária, de maneira transparente para o aplicativo, que passa a atuar consultando seu servidor. Porém, é dependente de aspectos lógicos de carga de acesso aos dados, o que a torna desvantajosa.

E em rede, em que a base de dados é fortemente acoplada. Sua vantagem é que cada aplicativo acessa apenas o servidor que dispõe dos seus dados, através do acesso à rede.

## GABARITO

1. B
2. B
3. A
4. C
5. A
6. B
7. D
8. C
9. A
10. D
11. C
12. A
13. C
14. B
15. B
16. A

## CONSIDERAÇÕES FINAIS

Prezados Alunos,

Chegamos, pois, ao final da aula introdutória de Banco de Dados! As videoaulas referentes ao conteúdo visto até aqui já estão gravadas e devidamente disponibilizadas na área do aluno. Espero que gostem!

Forte abraço, bons estudos e até breve! Que Deus abençoe!

Thiago Cavalcanti

## REFERÊNCIAS

Fiz uma lista com alguns livros que são referências do conteúdo apresentado na aula caso você queria se aprofundar um pouco.

1. Fundamentals of Database Systems - Ramez Elmasri, Sham Navathe Addison-Wesley, 2011 - Computers - 1172 pages
2. Introdução a sistemas de bancos de dados - By C. J. Date - Elsevier Brasil, 2004 - 865 pages
3. Sistema de Banco de Dados - Abraham Silberschatz, Henry F. Korth, S. Sudarshan - Editora: ELSEVIER BRASIL

THIAGO CAVALCANTI
PROFESSOR

# ESSA LEI TODO MUNDO CONHECE:
# PIRATARIA É CRIME.

**Mas é sempre bom revisar o porquê e como você pode ser prejudicado com essa prática.**

1. 

Professor investe seu tempo para elaborar os cursos e o site os coloca à venda.

2. 

Pirata divulga ilicitamente (grupos de rateio), utilizando-se do anonimato, nomes falsos ou laranjas (geralmente o pirata se anuncia como formador de "grupos solidários" de rateio que não visam lucro).

3. 

Pirata cria alunos fake praticando falsidade ideológica, comprando cursos do site em nome de pessoas aleatórias (usando nome, CPF, endereço e telefone de terceiros sem autorização).

4. 

Pirata compra, muitas vezes, clonando cartões de crédito (por vezes o sistema anti-fraude não consegue identificar o golpe a tempo).

5. 

Pirata fere os Termos de Uso, adultera as aulas e retira a identificação dos arquivos PDF (justamente porque a atividade é ilegal e ele não quer que seus fakes sejam identificados).

6. 

Pirata revende as aulas protegidas por direitos autorais, praticando concorrência desleal e em flagrante desrespeito à Lei de Direitos Autorais (Lei 9.610/98).

7. 

Concurseiro(a) desinformado participa de rateio, achando que nada disso está acontecendo e esperando se tornar servidor público para exigir o cumprimento das leis.

8. 

O professor que elaborou o curso não ganha nada, o site não recebe nada, e a pessoa que praticou todos os ilícitos anteriores (pirata) fica com o lucro.



**Deixando de lado esse mar de sujeira, aproveitamos para agradecer a todos que adquirem os cursos honestamente e permitem que o site continue existindo.**