

## Apresentação

Banco de dados é um conjunto de valores e arquivos que se relacionam entre si e formam um cadastro de pessoas, vendas, produtos, agenda, etc. Uma planilha ou uma ficha cadastral podem ser exemplos de cadastros. O banco de dados guarda todos os cadastros no formato virtual e os disponibiliza para as aplicações consultarem e emitirem relatórios, realizarem vendas, etc. Essas aplicações referem-se aos programas de software ou sistemas que utilizam os dados armazenados no banco de dados para realizar diversas funções. Elas podem ser desenvolvidas para uma variedade de propósitos, como gerenciamento de informações, automação de processos, geração de relatórios, análise de dados, entre outros.

Além disso, os bancos de dados oferecem recursos para garantir a integridade, segurança e eficiência dos dados, incluindo técnicas de *backup*, criptografia e controle de acesso. Eles são fundamentais para o funcionamento de muitas tecnologias modernas e desempenham papel crucial em inúmeras operações comerciais e pessoais do dia a dia.

Nesta Unidade de Aprendizagem, você vai estudar a evolução dos bancos de dados em virtude das necessidades impostas por seus utilizadores ou para solucionar problemas da sociedade. Adicionalmente, vai ver, em uma escala de tempo, os principais marcos da história dos bancos de dados com suas características e particularidades. Por fim, vai conhecer o início de tudo, o que deu origem aos bancos de dados como são conhecidos hoje em dia.

Bons estudos.

**Ao final desta Unidade de Aprendizagem, você deve apresentar os seguintes aprendizados:**

- Identificar a evolução dos bancos de dados.
- Reconhecer a ordem cronológica dos bancos de dados.
- Relacionar as origens dos bancos de dados.

# Infográfico

---

Ao comparar um banco de dados relacional (SQL) com um banco de dados não relacional (NoSQL), é importante entender as características de cada um e como elas se alinham com os requisitos do projeto. Os bancos de dados relacionais são baseados em um modelo de tabelas inter-relacionadas, em que os dados são estruturados e organizados conforme um esquema predefinido.

Por outro lado, os bancos de dados NoSQL são mais flexíveis em termos de esquema e são projetados para lidar com grandes volumes de dados não estruturados ou semiestruturados, usando modelos como documentos, grafos ou chave-valor.

Neste Infográfico, explore o comparativo entre um banco de dados relacional e um banco de dados não relacional: SQL vs. NoSQL. Esse conteúdo é muito útil para efetuar uma escolha de uso ou, ainda, definir a compra de um novo sistema.

# SQL E NOSQL: EVOLUÇÃO E COMPARAÇÃO

Em um mercado cada vez mais diversificado de sistemas de gerenciamento de banco de dados (SGBD), a escolha entre SQL e NoSQL tornou-se uma decisão crucial para muitas empresas e desenvolvedores. SQL e NoSQL são duas abordagens distintas para o gerenciamento de banco de dados, cada uma com características e vantagens específicas.

Este infográfico oferece uma comparação histórica entre essas duas abordagens, destacando suas origens, evolução e aplicações típicas, a fim de fornecer uma visão abrangente para ajudar na seleção do banco de dados mais adequado às necessidades específicas de cada projeto ou organização. Saiba mais a seguir.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

1974

2009

## SQL

Criado em 1974, o SQL introduziu um modelo de banco de dados relacional que revolucionou o armazenamento e a recuperação de dados.

Conheça algumas de suas principais características:

- > **Orientado a relacionamentos** (banco de dados relacional): organiza os dados em tabelas que podem ser conectadas por chaves estrangeiras, garantindo integridade e consistência.
- > **Menor velocidade:** em geral, apresenta uma velocidade de processamento mais lenta devido à complexidade das consultas e à necessidade de realizar junções entre tabelas.
- > **Controle:** oferece um alto nível de controle sobre a estrutura e a integridade dos dados, o que o torna adequado para aplicações que requerem rigor e precisão.
- > **Alta adesão:** tem sido a escolha padrão de empresas e desenvolvedores para uma ampla variedade de aplicações devido à sua estabilidade, robustez e ao seu suporte de longa data.

Embora tenha sido dominante por décadas, está enfrentando desafios com o surgimento de tecnologias e requisitos de escalabilidade e flexibilidade.

## NOSQL

Lançado em 2009, o NoSQL emergiu como uma resposta aos desafios enfrentados pelos bancos de dados relacionais em lidar com grandes volumes de dados não estruturados.

Conheça algumas de suas principais características:

- > **Orientado a documentos:** armazena os dados em documentos flexíveis, como JSON ou XML, permitindo uma modelagem mais dinâmica e adaptável.
- > **Maior velocidade:** oferece uma velocidade de processamento superior, especialmente para consultas simples, devido à sua estrutura simplificada e descentralizada.
- > **Escalabilidade:** projetado para escalar horizontalmente, permitindo lidar facilmente com grandes volumes de dados distribuídos em vários servidores.
- > **Baixa adesão:** embora esteja ganhando terreno, ainda enfrenta resistência em certos setores devido à falta de padronização e ao menor histórico de uso.

Considerado por muitos como o futuro do armazenamento de dados, especialmente para aplicações que exigem escalabilidade, flexibilidade e velocidade.

A decisão entre SQL e NoSQL deve ser cuidadosamente ponderada, levando em consideração diversos fatores, como requisitos de desempenho, complexidade dos dados, escalabilidade e experiência da equipe. Em última análise, ambas as abordagens têm lugar no mercado de banco de dados e podem ser utilizadas com sucesso em diferentes contextos, dependendo das necessidades de cada projeto.

Banco de dados relacional é um sistema de armazenamento estruturado de dados em que as informações são organizadas em tabelas com relações definidas, oferecendo consistência e integridade a elas.

Já o banco de dados NoSQL representa uma abordagem flexível para armazenar dados não estruturados ou semiestruturados, permitindo escalabilidade e desempenho elevados. No vídeo interativo a seguir, você verá um comparativo desses dois modelos.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

# Conteúdo do Livro

---

A constante evolução dos bancos de dados tem sido marcada pelo contínuo aprimoramento em termos de segurança e velocidade no armazenamento e na recuperação de informações.

Conforme destacado por Korth, Silberschatz e Sudarshan (2012), um banco de dados é essencialmente uma coleção de dados inter-relacionados, representando informações sobre um domínio específico. Essa definição destaca a importância crítica dos bancos de dados como ferramentas fundamentais para a organização e o acesso eficiente aos dados, desempenhando papel central em uma ampla gama de aplicações e setores.

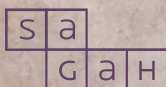
Além disso, a evolução dos bancos de dados tem sido impulsionada por avanços tecnológicos contínuos, como a computação em nuvem e o processamento de dados em tempo real, que têm possibilitado novas formas de armazenamento, análise e utilização dos dados. Essas inovações têm ampliado o escopo e a capacidade dos bancos de dados, permitindo que as organizações extraiam *insights* valiosos e tomem decisões estratégicas com base em dados mais precisos e oportunos.

No capítulo **Evolução dos bancos de dados**, base teórica desta Unidade de Aprendizagem, você vai estudar a origem e de que maneira evoluíram os bancos de dados.

Boa leitura.

# MODELAGEM E DESENVOLVIMENTO DE BANCO DE DADOS

Fabício Felipe Meleto Barboza



SOLUÇÕES  
EDUCACIONAIS  
INTEGRADAS



# Evolução dos bancos de dados

## Objetivos de aprendizagem

Ao final deste texto, você deve apresentar os seguintes aprendizados:

- Identificar a evolução dos bancos de dados.
- Reconhecer a ordem cronológica dos bancos de dados.
- Explicar as origens dos bancos de dados.

## Introdução

Um banco de dados é um conjunto de valores e arquivos que se relacionam entre si e demonstram um cadastro de pessoas, vendas, produtos, agendas, etc. Por exemplo, uma planilha ou uma ficha cadastral podem ser exemplos de cadastros. O banco de dados guarda todos esses cadastros em formato virtual e os disponibiliza para as aplicações consultarem e emitirem relatórios, realizarem vendas, etc.

Neste capítulo, você vai estudar sobre a evolução dos bancos de dados decorrente das necessidades impostas por seus utilizadores e para solucionar problemas da sociedade. Adicionalmente, você verá, em uma escala de tempo, os principais marcos da história dos bancos de dados com suas características e particularidades. Por fim, irá conhecer o início de tudo, isto é, o que deu origem aos bancos de dados como são conhecidos hoje em dia.

## A evolução dos bancos de dados

A evolução dos bancos de dados foi e continua sendo de constante melhoria, trazendo mais segurança e rapidez para as informações ou dados nele salvas. Para Korth, Silberschatz e Sudarshan (2012), um banco de dados “[...] é uma coleção de dados inter-relacionados, representando informações sobre um domínio específico”.

Pense em uma situação hipotética em que você tenha uma loja de sapatos em um shopping. Você está verificando a implantação de um sistema para controlar todas as suas vendas e também as compras de reposição de estoque, de maneira que todas as informações da sua loja estarão salvas em um sistema no computador da loja que, obviamente, utiliza um banco de dados.

Agora imagine se o vendedor do sistema pedisse pra você escolher entre 1) um sistema rápido, que tira relatórios gigantescos em menos de um segundo, processa vendas também em menos de um segundo do relógio, mas, por outro lado, não garante que a informação está íntegra e consistente, e 2) um banco de dados que garante a integridade e a consistência das informações, mas demora dez segundos para processar alguma venda ou um minuto para criar o relatório complexo solicitado. Qual você, dono da loja, escolheria?



### Fique atento

Lembre-se de que a evolução em sistemas de tecnologia da informação é importante, mas a segurança sempre deve vir em primeiro lugar para que, depois, possamos pensar em estabilidade, funcionalidade e velocidade.

Uma tendência da evolução comum é sempre mirar e colocar o foco e as metas em realizar a ação com mais e mais rapidez, cada vez provendo maior velocidade e resultados instantâneos para o usuário do sistema. Porém, ao decidir pela evolução de um sistema de banco de dados, é necessário pensar em outros fatores além da velocidade. Com toda a certeza, a velocidade é importante e sempre levada em consideração, mas os outros pilares precisam ser garantidos para toda transação de um banco de dados. Esses pilares são:

- Atomicidade;
- Isolamento;
- Consistência;
- Durabilidade.

Essas quatro características são garantidas para cada transação realizada no banco de dados, independentemente de quem realizou o comando: o próprio sistema, o usuário final ou ainda alguma rotina automática.



Mas o que seria uma transação em um sistema de banco de dados? Uma transação para um sistema de gerenciamento de banco de dados (SGBD) é toda e qualquer atividade que o próprio sistema de gerenciamento de banco de dados executa após o usuário ter uma interação com o banco.

Assim, qualquer atividade provocada pelo usuário via interface do sistema dispara uma transação ou uma cadeia de transações no banco de dados que precisam estar garantidas e ter aderência a esses pilares. Consequentemente, essa é a regra de ouro dos bancos de dados.



### Fique atento

A regra de ouro dos sistemas de gerenciamento de bancos de dados é a garantia de que todos os pilares das transações sejam assegurados e respeitados em cada uma de suas transações internas.

Sabendo disso, agora, é necessário entender o que cada um desses pilares representa, defende e significa para os bancos de dados.

## Atomicidade

Para entender o pilar da atomicidade, fazemos uma analogia à expressão “08 ou 80”, ou seja, ou finaliza com sucesso ou tudo é abortado. Portanto, ou o SGBD conclui todas as ações em cascata para uma transação (*commit*) ou ele retorna ao estado anterior da transação (*rollback*).

Assim, a atomicidade é a responsável por garantir que ou uma transação é gravada com sucesso e garantida ou é realizado o *rollback* de toda a transação para que o banco de dados retorne ao estado anterior daquela transação. Nunca um sistema de gerenciamento de banco de dados pode permitir que uma transação seja executada pela metade e fique sem *rollback*, pois isso faz a consistência do banco de dados ficar quebrada.



### Saiba mais

Atomicidade é a regra do “ou tudo ou nada”. Ou toda a transação é salva com sucesso no banco ou toda a transação é descartada.

## Isolamento

Este pilar representa a independência de cada transação no sistema de gerenciamento de banco de dados. Cada transação é única e independente, o que faz com que duas transações que alterem o mesmo valor de uma tabela não entrem em conflito. Toda transação é uma engrenagem no sistema maior, que seria o SGBD.

Conforme o sistema tem mais usuários trabalhando nele, mais esta regra se torna verdadeira e necessária. Imagine um *e-commerce* que realiza venda de enfeites natalinos. Quando chegar a época de procura por esses enfeites, o fluxo de pessoas utilizando o sistema irá aumentar consideravelmente. Agora imagine que existe apenas um item de Papai Noel em estoque e dois clientes realizam a compra desse item no mesmo segundo; como resolver? O pilar de isolamento garante que a transação que for processada em primeiro lugar pelo sistema de gerenciamento de banco de dados leve a compra e que a segunda pessoa receba um alerta de que o produto ficou fora de estoque.

No exemplo, duas transações estavam tentando inserir dados referentes ao mesmo valor único, que seria a quantidade de estoque do produto. Por tratar-se de duas transações isoladas e independentes, o sistema de gerenciamento de banco de dados rapidamente resolve essa questão, de modo que tal responsabilidade não fica para a equipe de desenvolvimento.



### Saiba mais

Isolamento é a garantia de que as transações não terão disputa para edição ou escrita em um determinado dado ou valor.

## Consistência

As regras, em sua totalidade, **sempre** devem ser respeitadas para que o sistema de gerenciamento de banco de dados tenha atendido suas características de base. Isso inclui, por exemplo, que o tipo de valor inserido seja correto em relação ao tipo do seu atributo (VARCHAR, INT, DATE, etc.).

Assim, em face da consistência, um campo do tipo *Integer* nunca poderá receber caracteres alfabéticos. Essa separação se dá para a criação de buscas e resultados mais rápidos em campos do tipo numérico. Computadores só sabem interpretar números, de modo que campos não numéricos por natureza exigem

mais processamento do computador, que precisa transformar a informação alfabética salva em códigos e linguagem de máquinas que são representados por números para conseguir trabalhar.



### Saiba mais

Consistência é ter o dado certo no local esperado. Um campo do tipo *Integer* sempre deverá conter um valor de número inteiro.

## Durabilidade

Quando uma transação é finalizada, seus dados estão a salvo de qualquer modificação. Somente outra transação pode modificar os dados. Assim, os dados ficam protegidos!

Imagine que a sua compra foi finalizada com sucesso e, quando você vai buscar seu histórico de compras, ela não aparece. Ruim, correto? A durabilidade garante que a informação de sua compra só seja modificada por outra transação e, dessa forma, consegue-se rastrear as requisições e buscar o que ou quem fez tal atividade.

O próprio sistema de gerenciamento de banco de dados não realiza nenhuma modificação após a conclusão de todas as transações pendentes. Portanto, qualquer modificação de um valor do banco de dados será de origem do usuário ou do sistema.

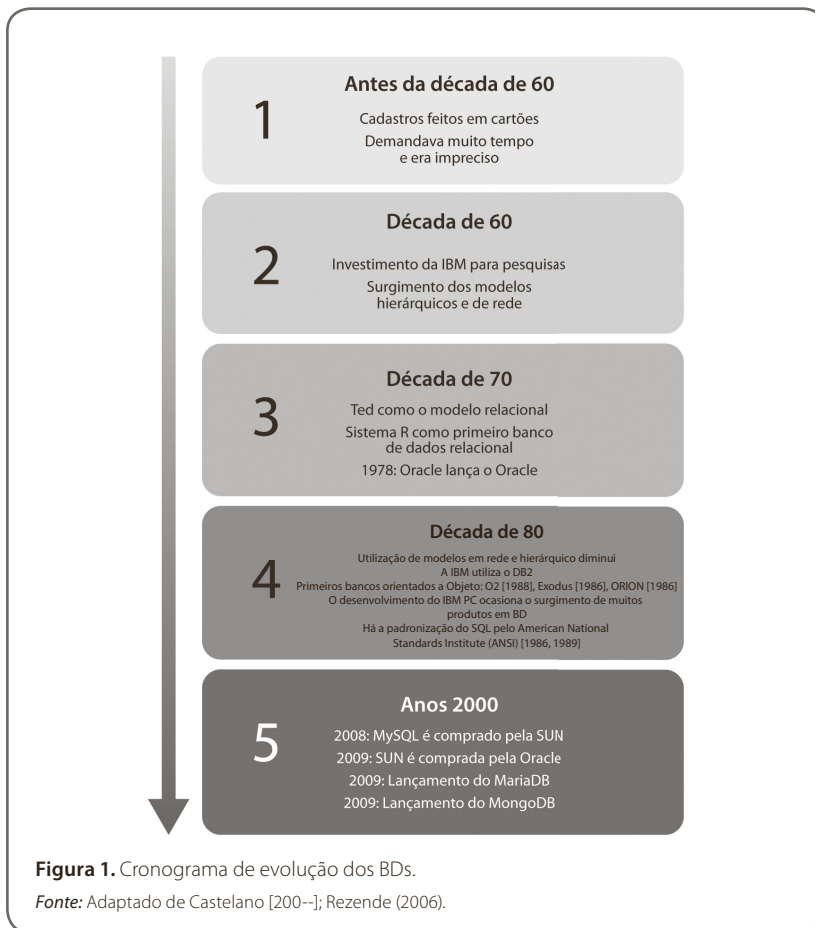


### Saiba mais

Durabilidade é a proteção dos dados contra qualquer coisa que não seja uma nova transação do banco de dados.

## Cronologia dos bancos de dados

Há divergência na ordem cronológica do surgimento das versões dos bancos de dados, exceto as versões mais comumente utilizadas. Observe a Figura 1, que representa as etapas mais importantes da evolução dos bancos de dados.



Com base na Figura 1, percebe-se grande concentração de lançamentos e popularização dos bancos de dados no período compreendido entre os anos 1980 até os 2000. Essa intensa movimentação se deu em virtude da informatização das empresas, que aconteceu ao longo desses anos. Cada empresa precisou incrementar seu parque tecnológico para que atendesse seus objetivos internos, como redução de custo, melhora do valor da entrega ao cliente, ou também objetivos externos, como estar de acordo com uma nova lei, receber auditoria de órgãos competentes ou, ainda, pela concorrência, que está em franca evolução e em busca por melhores resultados a todo custo.

Antes da década de 1980, temos o estágio embrionário dos bancos de dados. Sua evolução estava ocorrendo, mas era centralizada em forma de pesquisas

e experimentos. Até porque, quando fossem apresentados para as empresas e para o público geral, era necessário confiar no produto oferecido ao custo de milhões de dólares em pesquisas.

Após os anos 2000, a evolução continua até os dias atuais, mas de forma menos acelerada, já que o foco, atualmente, é em estabilidade e segurança cada vez maiores. Partindo desse princípio, temos o surgimento de bancos de dados NoSQL, que fazem frente aos tradicionais bancos de dados relacionais SQL.

Retome a Figura 1 e atente-se para o quadrante “Anos 2000”, no qual você verá que foi nesse período que surgiu o MongoDB, banco de dados NoSQL (*Not only SQL*) e com grande poder de processamento. Aí você pode se perguntar: “mas se ele é melhor, por que todos não migram pra ele?”. A resposta é direta: custo e foco do projeto.

O banco de dados NoSQL não veio para matar os bancos de dados relacionais como muitos imaginavam em seu lançamento, mas para complementar e oferecer uma opção a mais para as empresas e desenvolvedores segundo Guedes (2017).

Dessa forma, podemos exemplificar o uso de cada um dos tipos de bancos de dados de forma mais didática: caso a aplicação tenha relacionamentos entre as entidades e o volume de informações não for gigantesco, os bancos de dados relacionais tradicionais atenderão muito bem a sua demanda por um sistema.

Agora, caso você tenha um sistema de bancos de dados grandioso, com muitos valores sendo escritos e lidos ao mesmo tempo, a indicação é um banco de dados do tipo NoSQL para que o seu sistema ganhe em desempenho e não frustre os usuários com telas de “carregando”, erros de *timeout* de conexão ou ainda a negação de serviço por falta de recursos ao processar uma consulta gigante do gerente que trabalha remoto, impedindo de abrir a agenda de reuniões de outro funcionário.

Lembre-se também de que, por ser uma tecnologia mais nova, o custo para manter um banco de dados NoSQL é grande, até pela escassez de mão de obra qualificada no mercado de trabalho. Portanto, mensure esse custo ao tomar uma decisão sobre qual tipo de sistema de gerenciamento de bancos de dados utilizar no projeto.

## Origens dos bancos de dados

Antigamente, as empresas produziam e mantinham cadastros de tudo que fosse necessário em fichas e papéis armazenados em um ou vários arquivos

dentro de uma ou várias salas. Portanto, cliente, fornecedor, colaborador, agendas de compromisso, agendas telefônicas, inventário, controle de estoque, etc. estavam em um papel, em algum lugar dentro de algum arquivo.

Obviamente, organizar e manter tudo atualizado era uma tarefa complexa demais, pois envolvia um esforço muito grande para buscar informações básicas e que deveriam estar acessíveis para qualquer colaborador de forma instantânea.

Imagine que é necessário buscar todos os registros de ponto de entrada e saída de um colaborador que ficou durante vinte anos na empresa pois ele entrou com uma ação trabalhista, por exemplo. Agora imagine essa mesma situação e com um prazo de cinco dias para entregar toda essa informação ao Ministério do Trabalho. Impossível, certo?

Além disso, as informações contidas nesses papéis ficavam facilmente obsoletas em virtude da demora em atualizar o registro. Portanto, nada confiável.

Imagine a probabilidade de um desastre na sala de arquivos, como um cano de água rompido que encharque todos os registros da empresa. Com certeza, a empresa em questão teria sérios problemas ou, em um caso mais extremo, mas não descartado, até mesmo fecharia as portas se algo dessa magnitude acontecesse.

Pode-se ir mais a fundo e imaginar que algum funcionário espião ou insatisfeito com a empresa roube a informação de um projeto que iria ser lançado no próximo ano e a venda para a concorrência. Todo o esforço empregado na pesquisa e no desenvolvimento daquele projeto foi praticamente em vão, pois o elemento surpresa, para pegar o mercado no ponto certo e os concorrentes fiquem desequilibrados, foi perdido.



### Saiba mais

Sempre que um problema é detectado, ele deve ser resolvido logo por meio do fomento de ideias para o seu contorno ou para a criação de solução definitiva.

Pense que os bancos de dados só foram inventados porque existiu uma necessidade latente de diminuir despesas e melhorar a confiança nas informações salvas pela empresa com o passar dos anos.

Dessa forma, havia um problema grande e generalizado em todas as corporações em que registros eram feitos em papéis e armazenados em um lugar qualquer, muitas vezes sem a devida importância para o dado, para o registro ali contido. Como grandes problemas sugerem soluções altamente lucrativas, iniciou-se pesquisas que apresentassem uma solução para todo esse caos vivenciado em cada pequeno, médio ou grande escritório ou empresa no mundo.

Na década de 1960, surge uma solução para esse caos: o banco de dados. Vários modelos foram apresentados à IBM, financiadora do projeto, e, entre os modelos propostos, estavam o hierárquico e o de rede.

Apesar disso, nada muito usual ou com uma viabilidade de escala foi apresentado, o que fez os projetos serem engavetados.

Segundo Alves (2013), já na década de 1970, mais precisamente no mês de junho, o então pesquisador da IBM Edgar Frank “Ted” Codd apresentou um artigo que mudaria a história dos modelos de bancos de dados para sempre – ele estava propondo o modelo relacional de bancos de dados. Em seu artigo *Relational Model of Data for Large Shared Data Banks*, Ted descrevia como usuários leigos conseguiriam extrair os dados que lhes eram importantes.

No raciocínio de Alves (2013), vemos que, já no fim da década de 1970, surge o Sistema R, um sistema baseado nas ideias de Ted. Junto ao lançamento do Sistema R, veio a linguagem SQL, acrônimo para *Structured Query Language*, a linguagem que dominou o mercado. O Sistema R não foi bem comercialmente e acabou sendo abandonado, mas serviu, com toda a certeza, de inspiração e base para muitos dos bancos de dados que temos nos dias de hoje.

Finalizando seu pensamento, Alves (2013) diz que, na década de 1980, ocorrem dois marcos importantes na linha cronológica dos bancos de dados: o lançamento do Oracle 2 pela própria Oracle e, também, o lançamento do SQL/DS pela IBM, conhecido por DB2 atualmente.

Portanto, temos aí a origem dos bancos de dados. Com o passar de mais anos e décadas, os sistemas de gerenciamento de bancos de dados foram evoluindo segundo a necessidade dos clientes, no caso de sistemas pagos, ou pela produção da comunidade, no caso de sistemas *opensource*.

Aliás, uma curiosidade que também permeia este cenário de evolução: sempre que a comunidade dos sistemas de bancos de dados *opensource* lança algo inovador, as empresas que têm bancos de dados licenciados correm atrás e vice-versa. Com isso, muitas funcionalidades que estão presentes em sistemas de bancos de dados *opensource* também se tornam presentes nos sistemas proprietários, assim como o inverso.

A aplicação de cada sistema de banco de dados se dará pela escolha crucial de quem irá desenvolver baseado no orçamento e conhecimento daquela tecnologia de banco de dados específica para otimizar o tempo do projeto e também o seu custo.



### Fique atento

Lembre-se de quem fez a pesquisa para obter os bancos de dados relacionais que conhecemos hoje: Edgar Frank "Ted" Codd, pesquisador da IBM na época e que lançou o artigo intitulado *Relational Model of Data for Large Shared Data Banks*. Ted é considerado o pai dos bancos de dados relacionais.



### Referências

ALVES, G. F. O. *A história dos bancos de dados*. 01 abr. 2013. Disponível em: <<https://dicas-deprogramacao.com.br/a-historia-dos-bancos-de-dados/>>. Acesso em: 21 maio 2018.

CASTELANO, C. R. *História dos bancos de dados*. [20--] Disponível em: <http://castelano.com.br/site/aulas/bd/Aula%2001%20-%20Introdu%C3%A7%C3%A3o.pdf>. Acesso em: 23 jun. 2021.

GUEDES, M. *SQL vs NoSQL, qual usar?* 19 jul. 2017. Disponível em: <<https://www.treina-web.com.br/blog/sql-vs-nosql-qual-usar/>>. Acesso em: 21 maio 2018.

KORTH, H. F.; SILBERSCHATZ, A.; SUDARSHAN, S. *Sistema de banco de dados*. 6. ed. Rio de Janeiro: Campus, 2012. Leituras recomendadas

CODD, E. F. A Relational Model of Data for Large Shared Data Banks. *Communications of the ACM*, v. 13, n. 06, June 1970.

REZENDE, R. *A história dos bancos de dados*. 2006. Disponível em: <https://www.dev-media.com.br/a-historia-dos-banco-de-dados/1678>. Acesso em: 23 jun. 2021.

REZENDE, R. *Conceitos fundamentais de banco de dados*. 2017. Disponível em: <<https://www.devmedia.com.br/conceitos-fundamentais-de-banco-de-dados/1649>>. Acesso em: 21 maio 2018.



Encerra aqui o trecho do livro disponibilizado para esta Unidade de Aprendizagem. Na Biblioteca Virtual da Instituição, você encontra a obra na íntegra.

Conteúdo:



SOLUÇÕES  
EDUCACIONAIS  
INTEGRADAS

# Dica do Professor

---

O MySQL é um dos bancos de dados mais amplamente utilizados no mundo, conhecido pela confiabilidade, pelo desempenho e pela facilidade de uso. Sua história remonta ao final da década de 1970, quando os fundadores Michael “Monty” Widenius e David Axmark começaram a desenvolver um sistema de gerenciamento de banco de dados que mais tarde se tornaria o MySQL.

Em 1995, eles lançaram a primeira versão pública do MySQL, marcando o início de uma jornada que o levaria a se tornar uma das principais soluções de banco de dados de código aberto disponíveis atualmente.

Nesta Dica do Professor, você vai ver a história do MySQL, um dos maiores bancos de dados existentes.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

# Exercícios

- 1) Com o avanço da tecnologia e o crescimento exponencial na geração de dados, a evolução dos bancos de dados tornou-se uma necessidade. Dos sistemas de bancos de dados relacionais tradicionais até as modernas soluções NoSQL, as organizações têm buscado maneiras eficientes de armazenar, gerenciar e analisar dados para obter *insights* valiosos e tomar decisões estratégicas informadas.

Uma empresa de comércio eletrônico está enfrentando dificuldades para lidar com o rápido crescimento de dados gerados por seus clientes. Eles estão considerando migrar de um banco de dados relacional para uma solução NoSQL a fim de melhorar o desempenho e a escalabilidade de sua plataforma.

Qual das seguintes alternativas representa uma vantagem significativa dos bancos de dados NoSQL em comparação com os bancos de dados relacionais SQL?

- A) Estrutura rígida de dados.
- B) Limitada escalabilidade horizontal.
- C) Complexidade na modelagem de dados.
- D) Flexibilidade de esquema e escalabilidade horizontal.
- E) Alta aderência a transações ACID.

- 2) Os sistemas de gerenciamento de banco de dados (SGBD) têm papel fundamental na organização e manipulação de dados em ambientes computacionais. Desde suas origens nos sistemas hierárquicos e em rede até os modernos sistemas relacionais e NoSQL, os SGBD evoluíram para atender às crescentes demandas de armazenamento, processamento e análise de dados em uma variedade de contextos.

Uma empresa de saúde está buscando um SGBD que possa lidar com a complexidade e o volume de dados de registros médicos de pacientes. Eles precisam de uma solução que garanta a integridade e a segurança dos dados ao mesmo tempo em que oferece flexibilidade para consultas e análises avançadas.

Considerando o estudo de caso da empresa de comércio eletrônico mencionada, qual das seguintes alternativas melhor justifica a necessidade de migrar de um banco de dados

relacional para uma solução NoSQL?

- A) Dificuldade em manter a integridade referencial dos dados.
  - B) Crescimento rápido e imprevisível de dados não estruturados.
  - C) Necessidade de suportar transações complexas e relacionamentos rígidos.
  - D) Requisitos de segurança e controle de acesso aos dados.
  - E) Prioridade na otimização de consultas e desempenho de transações.
- 3) O uso de bancos de dados NoSQL na ciência de dados tem se destacado devido à sua capacidade de lidar com grandes volumes de dados não estruturados e semiestruturados de maneira eficiente. Esses sistemas oferecem flexibilidade de esquema, escalabilidade horizontal e suporte a modelos de dados variados, tornando-os adequados para uma variedade de casos de uso em ciência de dados, como análise de *big data*, aprendizado de máquina e processamento de fluxos de dados em tempo real.

Para cada uma das seguintes afirmações sobre características dos bancos de dados NoSQL em aplicações de ciência de dados, marque V (verdadeiro) ou F (falso).

I. Os bancos de dados NoSQL não têm uma dependência estrita da linguagem SQL para consultas e manipulação de dados.

II. A escalabilidade vertical é uma característica comum dos bancos de dados NoSQL, permitindo que se expandam facilmente conforme a demanda de dados aumenta.

III. Os bancos de dados NoSQL são menos flexíveis em termos de modelagem de dados em comparação com os bancos de dados relacionais SQL.

IV. A capacidade de escalabilidade horizontal dos bancos de dados NoSQL é particularmente vantajosa para lidar com grandes volumes de dados gerados por aplicações de ciência de dados em tempo real.

A ordem correta é apresentada em:

- A) V, F, F, V.
- B) V, F, V, V.

C) F, V, V, F.

D) V, F, V, F.

E) F, F, V, V.

4) Transações em banco de dados são unidades de trabalho que realizam operações sobre os dados de forma consistente e durável. Essas transações devem seguir as propriedades ACID, que garantem atomicidade, consistência, isolamento e durabilidade. Sobre as propriedades ACID, analise os itens a seguir:

I. Atomicidade assegura que todas as operações de uma transação sejam concluídas com sucesso ou revertidas, caso ocorra algum erro.

II. Consistência garante que as operações de uma transação não tenham disputa com outras transações concorrentes, evitando interferências indesejadas.

III. Isolamento garante que o banco de dados obedeça as regras do SGBD em sua totalidade antes e depois da transação.

IV. Durabilidade garante que as alterações feitas por uma transação permaneçam armazenadas, mesmo em caso de falhas do sistema.

Está correto apenas o que se afirma em:

A) I e IV.

B) II e III.

C) I e III.

D) I e II.

E) II, III e IV.

5) No decorrer das últimas décadas, observamos um crescimento exponencial no desenvolvimento e na adoção de tecnologias de bancos de dados. No entanto, é possível identificar um período de maior atividade e inovação nesse campo durante as últimas duas décadas do século XX e o início do século XXI. Esse intervalo de tempo foi marcado por avanços significativos tanto em *hardware* quanto em *software*, possibilitando o surgimento de sistemas de gerenciamento de banco de dados cada vez mais poderosos e sofisticados.

Nesse período, testemunhamos o surgimento de conceitos fundamentais, como bancos de dados relacionais, linguagem SQL e modelos de dados NoSQL, que moldaram profundamente a paisagem tecnológica atual.

Considerando o contexto histórico apresentado no texto, qual foi um dos avanços significativos que moldaram a paisagem tecnológica atual durante as duas últimas décadas do século XX e o início do século XXI?

- A) O surgimento dos primeiros bancos de dados NoSQL.
- B) A popularização dos sistemas de bancos de dados hierárquicos.
- C) O desenvolvimento de *hardware* de baixo desempenho.
- D) A criação de modelos de dados totalmente estruturados.
- E) A introdução da linguagem de consulta SQL.

# Na prática

---

Os bancos de dados relacionais, baseados na linguagem SQL (*structured query language*, ou, em português, linguagem de consulta estruturada), são amplamente utilizados e seguem um modelo de dados tabular, organizado em tabelas com linhas e colunas. Por outro lado, os bancos de dados NoSQL, como o próprio nome sugere, adotam uma abordagem diferente, permitindo maior flexibilidade no esquema dos dados e muitas vezes empregando modelos de dados como documentos, grafos ou chave-valor.

Enquanto os bancos de dados SQL são conhecidos por sua estrutura rígida e esquema predefinido, os bancos de dados NoSQL oferecem uma abordagem mais flexível, permitindo que os desenvolvedores armazenem e manipulem dados de forma mais dinâmica. Além disso, os bancos de dados SQL são tradicionalmente escalonados verticalmente, o que significa que a adição de recursos é feita aumentando a capacidade do servidor, enquanto os bancos de dados NoSQL são projetados para escalabilidade horizontal, distribuindo os dados em vários servidores para lidar com cargas de trabalho crescentes.

Acompanhe, Na Prática, a comparação entre um banco de dados relacional SQL e um banco de dados do tipo NoSQL.



## Como consultar em bancos de dados SQL e NoSQL?

Pense que o banco de dados onde será executada a consulta possui uma tabela chamada "cliente\_vip" com os seguintes atributos:

- nome;
- CPF;
- endereço;
- bairro;
- CEP;
- telefone;
- cidade;
- idade.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

### Consulta I - Retornar nome, CPF, idade e bairro dos clientes com mais de 18 anos:

#### SQL

```
SELECT NOME, CPF, IDADE, BAIRRO  
FROM cliente_vip  
WHERE IDADE > 18;
```

#### NoSQL

```
db.cliente_vip.find({  
  "IDADE": {  
    "$gt": 18  
  }  
}, {  
  "NOME": 1,  
  "CPF": 1,  
  "IDADE": 1,  
  "BAIRRO": 1  
});
```

### Consulta II - Retornar telefone e CEP de todos os clientes:

#### SQL

```
SELECT telefone, cep  
FROM cliente_vip;
```

#### NoSQL

```
db.cliente_vip.find({}, {  
  "telefone": 1,  
  "cep": 1  
});
```

Nesses exemplos, você viu que o tamanho da consulta para o NoSQL ficou maior, porém, em ambos os casos, retornaram os mesmos valores das tabelas (para o SQL) ou dos objetos (para o NoSQL).

Conforme o tamanho do banco de dados aumenta, a *performance* do NoSQL acaba vencendo facilmente o SQL. No entanto, pode ser mais trabalhoso montar as consultas que você deseja.



# Saiba mais

---

Para ampliar o seu conhecimento a respeito desse assunto, veja abaixo as sugestões do professor:

## Seis motivos para usar bancos de dados NoSQL

Veja quais são as vantagens de usar bancos de dados NoSQL.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

## O novíssimo RDS da Amazon

Saiba mais sobre o RDS, serviço de banco de dados da Amazon preparado para abraçar o mundo.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

## Evolução dos bancos de dados e SGBD

No texto a seguir, você vai ler um pouco mais sobre a evolução dos bancos de dados.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

## Sete anos de MySQL em sete minutos

Veja a história de um colaborador da Oracle falando sobre o MySQL.



Aponte a câmera para o código e acesse o link do conteúdo ou clique no código para acessar.

## **Projeto, desenvolvimento de aplicações e administração de banco de dados**

Este livro aborda os principais conceitos, técnicas e práticas relacionados ao projeto, ao desenvolvimento e à administração de sistemas de banco de dados. Ele oferece uma visão ampla, que vai desde os fundamentos teóricos até as aplicações práticas, abrangendo tópicos como modelagem de dados, linguagens de consulta, otimização de desempenho, segurança e administração de sistemas de banco de dados. Em especial, o Capítulo 1 fala dos principais conceitos de banco de dados. Não deixe de ler as páginas 3 a 20.

**Conteúdo interativo disponível na plataforma de ensino!**