**Descomplicando**

**Bancos de Dados**

**para estudantes de TI**

1. **Relaxa, ninguém vira expert de uma hora pra outra.**
2. **Vai com calma! Foca no que tá na sua frente e para de encanar com o que você ainda não sabe.**
3. **Manda ver! Quanto mais você pratica, mais você arrasa. Faz umas mudanças pequenas e segue ajustando.**
4. **Cria um desafio pra você mesmo resolver. Aqui, o que importa não é o resultado, mas o rolê de chegar lá.**
5. **Agora, dá uma revisada geral e refaz os passos anteriores. Bora lapidar tudo!**
6. **Introdução aos Bancos de Dados**  
   1.1. O que é um banco de dados?  
   1.2. Importância dos bancos de dados em TI  
   1.3. Aplicações práticas e casos de uso
7. **História dos Bancos de Dados**  
   2.1. O surgimento dos primeiros bancos de dados  
   2.2. Evolução dos modelos de banco de dados: hierárquico, em rede e relacional  
   2.3. O impacto do modelo relacional no mercado
8. **SQL: A Linguagem dos Bancos de Dados Relacionais**  
   3.1. Origem e história do SQL  
   3.2. Por que o SQL se tornou o padrão?  
   3.3. Evolução e variações do SQL
9. **Sistemas de Gerenciamento de Banco de Dados (SGBD)**  
   4.1. O que é um SGBD?  
   4.2. Tipos de SGBDs: Relacional, NoSQL, NewSQL e outros  
   4.3. Principais SGBDs do mercado: MySQL, PostgreSQL, Oracle, SQL Server e MongoDB
10. **Modelagem de Dados**  
    5.1. Conceitos básicos: Entidade, Relacionamento e Atributo  
    5.2. Diagramas Entidade-Relacionamento (ER)  
    5.3. Boas práticas na modelagem de dados
11. **Normalização de Dados**  
    6.1. O que é normalização e por que ela é importante?  
    6.2. As formas normais (1ª, 2ª, 3ª e BCNF)  
    6.3. Desnormalização: quando e por quê?
12. **Introdução à Linguagem SQL**  
    7.1. Estruturas básicas: SELECT, INSERT, UPDATE e DELETE  
    7.2. Filtros e condições: WHERE, AND, OR, e operadores  
    7.3. Ordenação e agrupamento de dados: ORDER BY e GROUP BY
13. **Consultas SQL Avançadas**  
    8.1. Funções agregadas: COUNT, SUM, AVG, MAX e MIN  
    8.2. Joins: INNER JOIN, LEFT JOIN, RIGHT JOIN e FULL JOIN  
    8.3. Subconsultas e CTEs (Common Table Expressions)
14. **Integridade e Segurança dos Dados**  
    9.1. Integridade referencial e chaves estrangeiras  
    9.2. Controle de acesso e permissões  
    9.3. Criptografia de dados e boas práticas de segurança
15. **Tópicos Complementares**  
    10.1. Introdução ao NoSQL: quando e por que usar  
    10.2. Indexação e otimização de consultas  
    10.3. Backups e recuperação de dados
16. **Projetos Práticos**  
    11.1. Criando seu primeiro banco de dados relacional  
    11.2. Exercícios de modelagem de dados  
    11.3. Consultas SQL aplicadas
17. **Conclusão e Próximos Passos**  
    12.1. Revisão dos conceitos apresentados  
    12.2. Recursos para aprofundar os estudos  
    12.3. Considerações finais

Apresentação

*Simplificar essa coisa chamada de* ***Banco de Dados Relacional*** *de modo a melhorar a comunicação com estudantes, e assim permitir que nasça mais um bom profissional de T.I. no mercado.*

O objetivo deste livro não é torna-lo expert em banco de dados, é por outro lado, ser uma espécie de ***Alavanca de Arquimedes*** para o seu entendimento sobre Bancos de Dados.

***"Dê-me uma alavanca e um ponto de apoio, e eu moverei o mundo"*** - ***Arquimedes.***

Esse livro será a ***Alavanca***, o ***Ponto de Apoio*** aqui será a base tecnológica e científica sobre bancos de dados, e o ***Movimento do Mundo*** será a sua mente na direção dos conhecimentos básicos sobre Bancos de Dados.

Espero sinceramente ao fim deste livro você seja capaz de realizar pequenas modelagens de bancos de dados relacionais, entenda os princípios de normalização de dados, ao menos até a terceira forma normal, e seja capaz de implementar bancos de dados relacionais de complexidade básica, bem como manipular seus dados e suas estruturas.

**Pra que serve um Banco de Dados?**

Um banco de dados nada mais é doque um meio para armazenar e recuperar dados e informações de maneira satisfatória, não tem nenhuma importância do como fazer isso, nem mesmo tem a ver com computadores necessariamente.

**Exemplos:**

**O caderno de um aluno na escola:** Um caderno comum não tem nada de computacional nele, porem nele é possível armazenar registros e informações de forma organizada, atualizar essas informações conforme a necessidade e recupera-las sempre um pouquinho antes de uma prova, não tem nada haver com bancos de dados relacionais, mas é um tipo de banco de dados.

**Tabelas de Excel:** Esse é um tipo muito usado em muitas empresas e por muitas pessoas, aqui também é possível armazenar, atualizar e recuperar dados e informações relevantes para cada contexto.

**Bancos de Dados Relacionais:** Aqui temos o estado da arte no contexto de bancos de dados *– isso é claro na minha humilde opinião*, como nos outros exemplos, o principal objetivo e um banco de dados relacional é o mesmo que o de um simples caderno, inserir dados, atualiza-los e recupera-los sempre que necessário, porem aqui o objetivo e atender uma necessidade bem maior. Os bancos de dados relacionais são soluções computacionais muito sofisticadas que oferece ao sistema que o opera uma série de vantagens como, suporte a multiusuário, segurança no acesso e na manutenção de dados dentre outros.

A grande questão aqui é viabilidade e contexto na hora de escolher como guardar seus dados. Um aluno na quinta série do fundamental não precisa um Oracle DB para organizar sua matéria de Matemática na escola, assim como uma dona de casa pode muito bem organizar suas finanças usando somente planilhas de Excel. Porem se a dona de uma loja de médio porte optar por usar apenas planilhas de Excel é possível que tenha muitas dificuldades em sua operação uma vez que, clientes, fornecedores, produtos dentre outros podem oferecer uma sobrecarga de informação, nesse caso ERP conectado a um banco de dados relacional pode lhe oferecer muitas vantagens.

**SGBD – Sistema Gerenciador de Banco de Dados**

Considere agora que sua realidade de contexto lhe obrigue a usar um banco de dados computacional, ou seja, você, de alguma forma precisará usar um banco de dados que deverá interagir com uma aplicação web, ou mobile, ou desk top, ou ainda todas estas, como seria feito isso exatamente?

*Através de um SGBD, é ele que estará entre os aplicativos todos e os seus dados de contexto, aqueles que você precisa guardar, ler, atualizar e deletar.*

Os SGBD´s são sistemas complexos, dos quais não vamos nos aprofundar muito em seus sistemas de funcionamentos, vamos nos concentrar em suas características básicas e fundamentais.