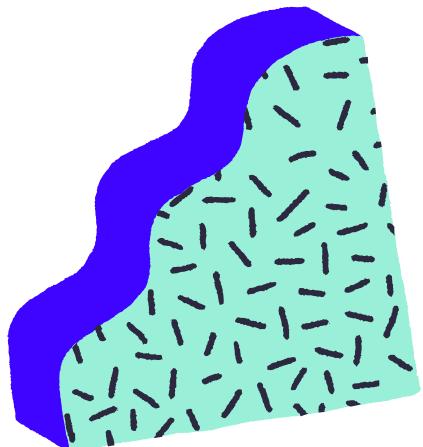


Banco de Dados Relacional



• • •
• • •





Francisco Douglas Lima Abreu



@franciscodougllas



francisco.abreu@fatec.sp.gov.br

Know me

Professor Universitário nos cursos de Tecnologia na Universidade de Mogi das Cruzes (UMC) e pela FATEC da Zona Leste

Desenvolvedor Web FullStack pela Quaestum Consultoria

MBA em Análise de Dados (2021) e Programação em Multiplataforma (2022) pela Faculdade Descomplica

Bacharel em Sistemas de Informação (2015), Mestre (2018) e Doutor (2022) em Engenharia Biomédica pela UMC



AULA 01

- EMENTA
 - SOBRE
 - METODOLOGIA
 - TÓPICOS DO SEMESTRE
 - INSTRUMENTO DE AVALIAÇÃO
 - AGENDA DE AVALIAÇÕES
 - REFERÊNCIA BIBLIOGRÁFICA
 - INTRODUÇÃO AO BANCO DE DADOS
 - TIME-LINE (ATÉ OS TEMPOS DE HOJE)
 - MERCADO DE T.I.
 - PROFISSIONAIS RELACIONADOS A ESSA DISCIPLINA
- 
- A thin, light blue wavy line that spans across the bottom right corner of the slide, extending from the bottom edge towards the right side.

EMENTA

SOBRE

- • •
- • •
- • •
- • •

Projeto e implementação de banco de dados relacionais. Consultas complexas com agrupamentos e subconsultas. Implementação de restrições de integridade. Criação de consultas utilizando visões. Aspectos de programação em ambiente de banco de dados com procedimentos armazenados, gatilhos e funções. Cópia de segurança e restauração de bancos de dados. Estruturas de índices. Processamento e otimização de consultas. Processamento de transações e controle de concorrência. Recuperação de falhas. Novas tecnologias aplicadas a banco de dados

METODOLOGIA

Aulas Expositivas. Aprendizagem Baseada em Projetos/Problemas. Gamificação, Estudo de Caso Real. Trabalhos Interdisciplinares desenvolvidos, seguindo o Manual de Projetos Interdisciplinares expedido pela CESU

TÓPICOS DO SEMESTRE

- Introdução a Banco de Dados
- Conceitos
- Modelagem de Dados
- Linguagem de Definição de Dados (DDL)
 - Criação, Alteração e Exclusão de Tabelas e Colunas
- Linguagem de Manipulação de Dados (DML)
 - Inserção, Atualização e Exclusão de Dados
- Linguagem de Consulta de Dados (DQL)
- Linguagem de Controle de Dados (DTL)

⋮
⋮
⋮
⋮
⋮

⋮ ⋮ ⋮ INSTRUMENTO DE AVALIAÇÃO



$$\mathbf{N1} = \text{AS} (60\%) + \text{AF} (40\%)$$

$$\mathbf{N2} = \text{PI} (30\%) + \text{AF} (10\%) + \text{AS} (60\%)$$

Avaliação Formativa (AF): São as atividades práticas realizadas em sala ou em casa(sem a necessidade de agendamento prévio, porém com prazo de entrega)

Avaliação Somativa(AS): São provas individuais, escritas e sem consulta, previamente agendadas;

Projeto Interdisciplinar (PI): Artefatos ou elementos que compões um Projeto de Banco de Dados Relacional, que serão realizados em aula e pós aula, com agendamento prévio para consultas de grupo e prazo de entrega.

- • • •
- • • •
- Nota do 1º Bimestre (N1)**
- Nota do 2º Bimestre (N2)**
- Nota do Exame (NE)**
- Média Final Semestral (MF)**

• • • **INSTRUMENTO DE AVALIAÇÃO**



$$\mathbf{MF} = \frac{\mathbf{N1} + \mathbf{N2}}{2}$$

EXAME

MF for ≤ 5.9 Será Aplicado a NE (100%), sendo:

$$\mathbf{MF} = \frac{(\mathbf{MF} + \mathbf{NE})}{2}$$

Nota mínima para aprovação na FATEC: 6.0

Nota do 1º Bimestre (N1)

Nota do 2º Bimestre (N2)

Nota do Exame (NE)

Média Final Semestral (MF)

• • •
• • •

• • • • **AGENDA DE AVALIAÇÕES**



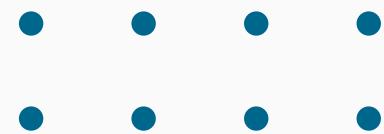
P1 :29/04/2023 às 8h:20

P2: 03/06/2023 às 8h:20

P3: 24/06/2023 às 8h:20

Duração das provas: 2h00

Prova do 1º Bimestre (P1)
Prova do 2º Bimestre (P2)
Prova do Exame (P3)

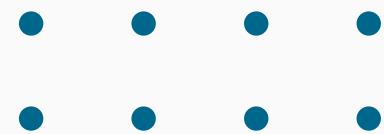


⋮ ⋮ ⋮ AGENDA DO PROJETO INTERD.



Os grupo apresentarão uma prévia sobre o seu projeto
20/05/2023 às 8h:00 - 11h30

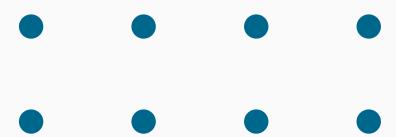
Duração para cada grupo: 10 minutos





PRESENÇA EM AULA

- Haverá uma tolerância de 15 (quinze) minutos para a entrada em sala, seja ela presencial ou virtual;
- Em aulas remotas (pelo MS Teams) sua presença será registrada em uma lista digital;
- Alunos que sejam chamados duas vezes e não interagirem oralmente ou por chat serão considerados ausentes;
- Pelo Regulamento, o Aluno poderá se ausentar por 25% das aulas sem reprovação.
- O aluno que ingressar na aula depois do período de tolerância terá falta relativa ao período de atraso (Lembrando que são dois módulos - 2 faltas para cada)

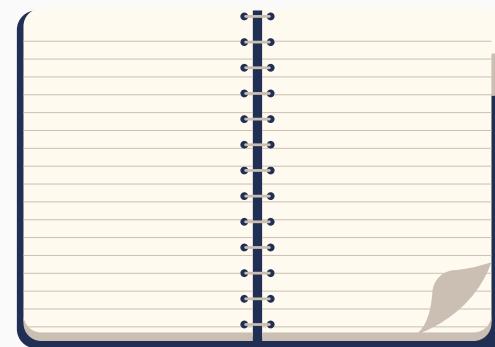
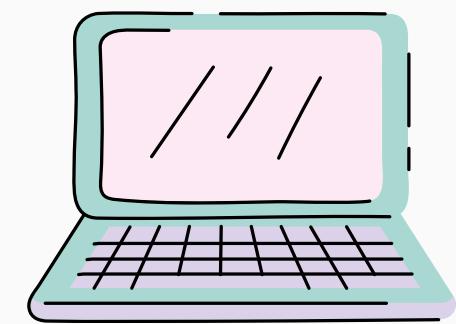




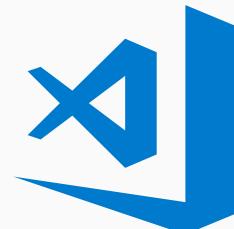
MATERIAL COMPLEMENTAR



Modo Remoto



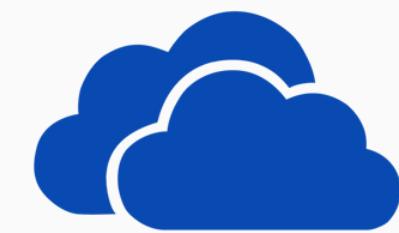
Opcional



Visual Studio Code



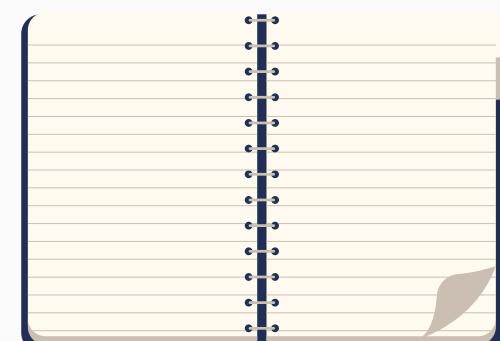
Opcional



OneDrive

Modo Presencial

Opcional



DISPONIBILIDADE DO MATERIAL



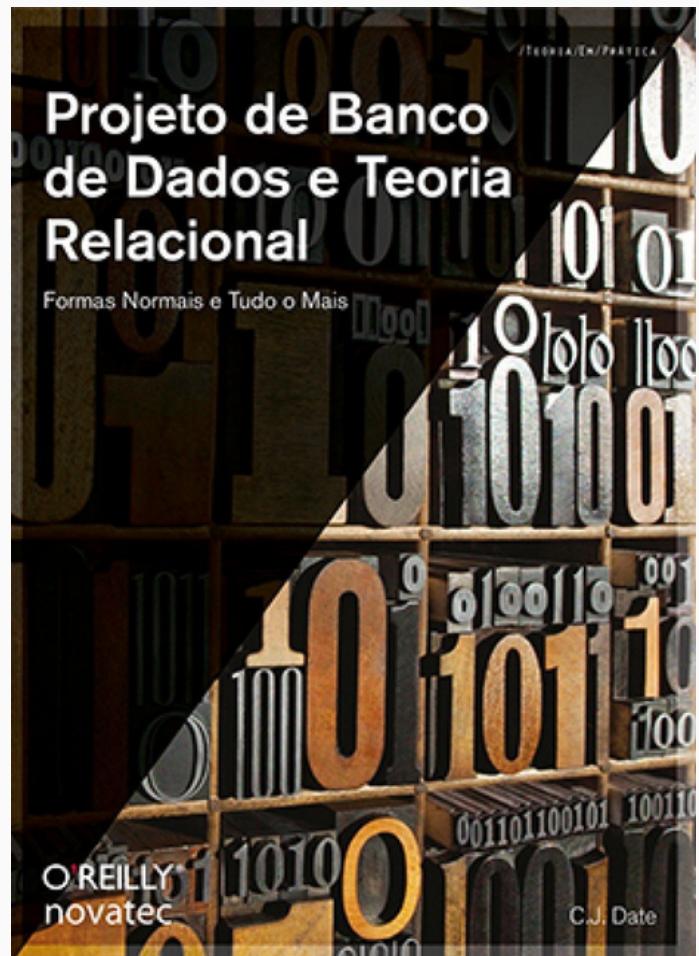
- Material Explicativo será disponibilizado no **final da aula**
- As Atividades em Aula serão disponibilizadas e entregues ao professor **no prazo estipulado**.
- As Atividades poderão ser entregues via Teams ou E-mail do Professor
- As **aulas NÃO serão gravadas**.

• • •
• • •

REFERÊNCIA BIBLIOGÁFICA

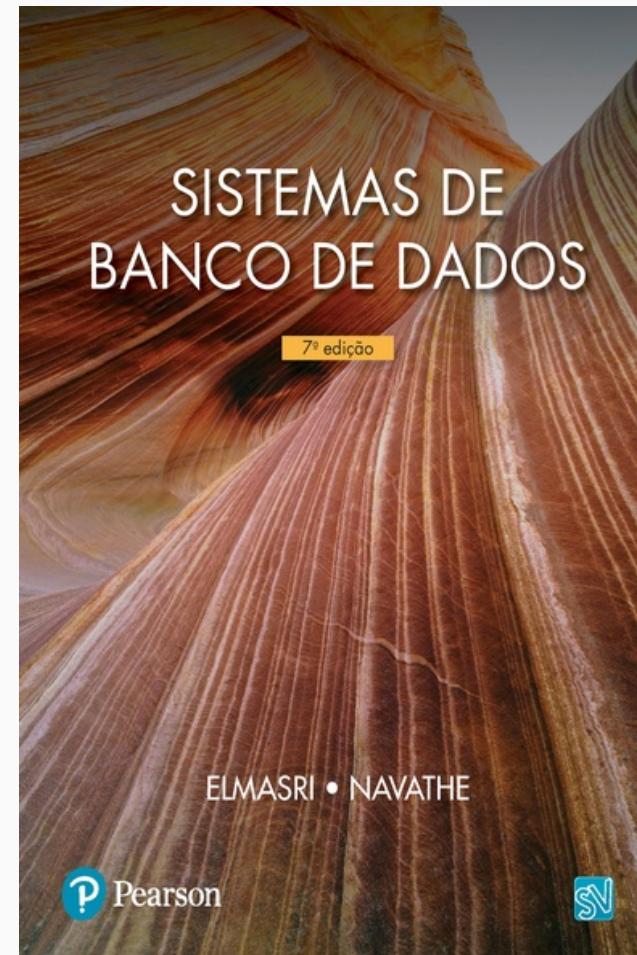


Bibliografia Básica:



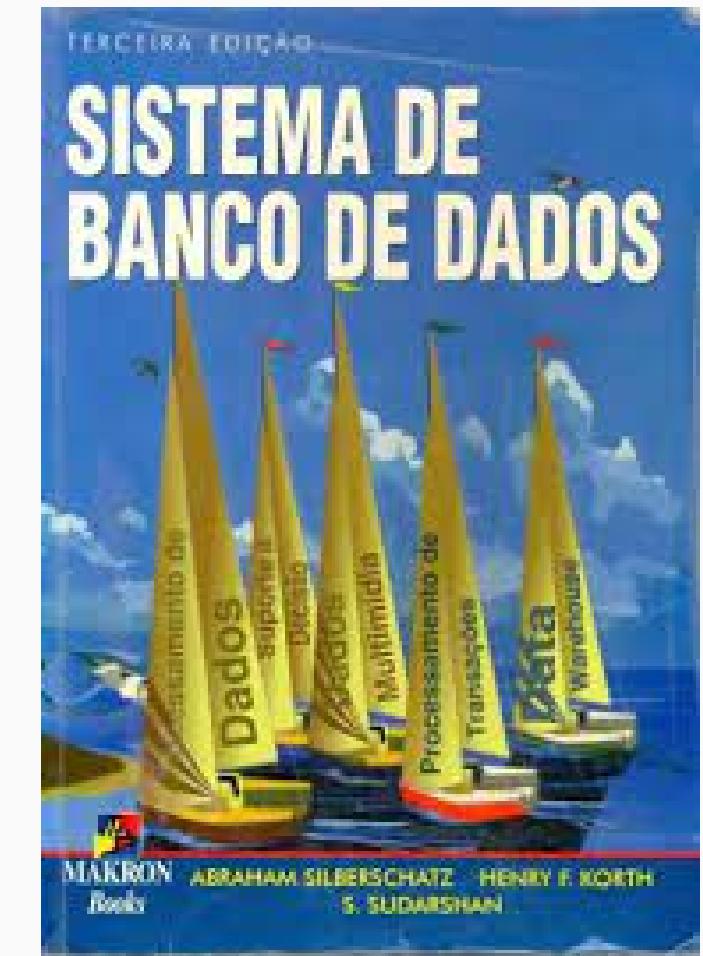
DATE, C. J.

Projeto de Banco de Dados e Teoria
Relacional: Formas Normais e Tudo o Mais.
São Paulo: Novatec, 2015.



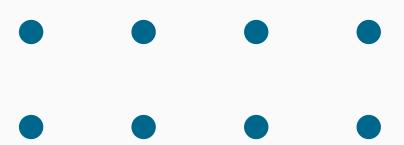
ELMASRI, R.; NAVATHE, S. B.

Sistemas de Banco de Dados. 7 ed. São Paulo:
Pearson, 2018.



**SILBERSCHATZ, Abraham; KORTH, Henry F;
SUDARSHAN, S.**

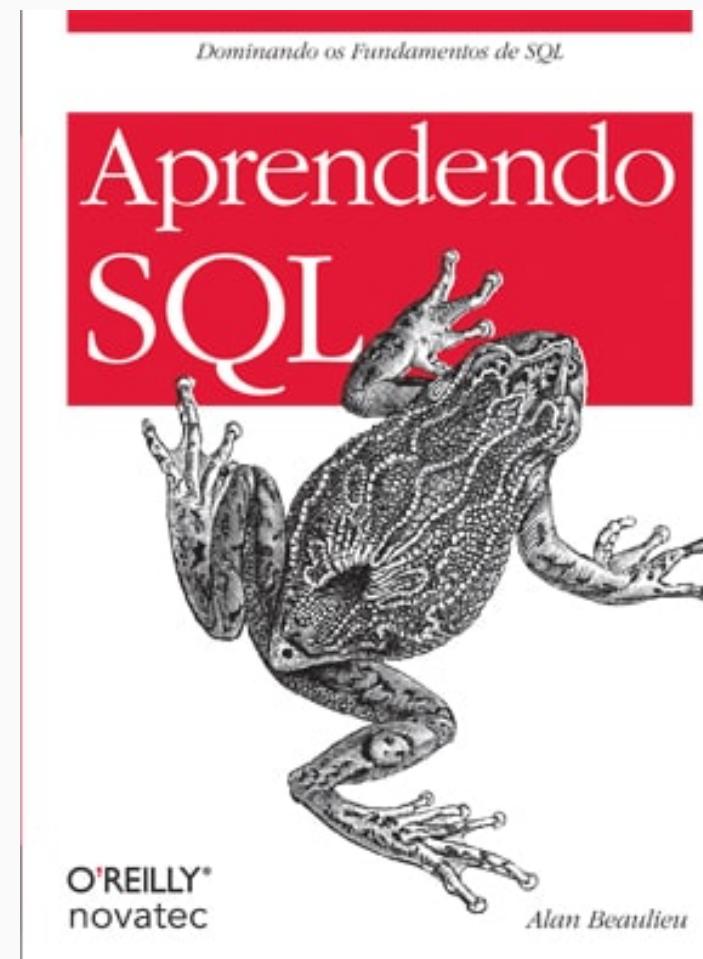
Sistema de banco de dados. 3^a edição. São
Paulo: Makron Books, 1999-2005



REFERÊNCIA BIBLIOGÁFICA



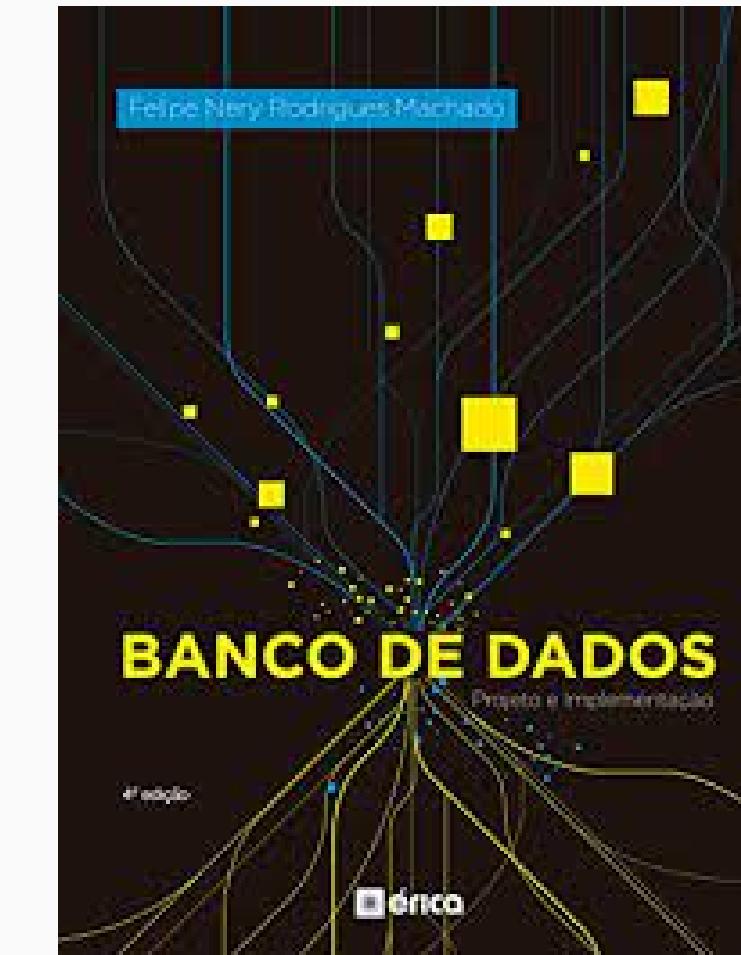
Bibliografia Complementar:



BEAULIEU, A.
Aprendendo SQL. São Paulo: Novatec, 2010.



GILLENSON, M. L.
Fundamentos de Sistemas de Gerência de Banco de Dados. Rio de Janeiro: LTC, 2006.



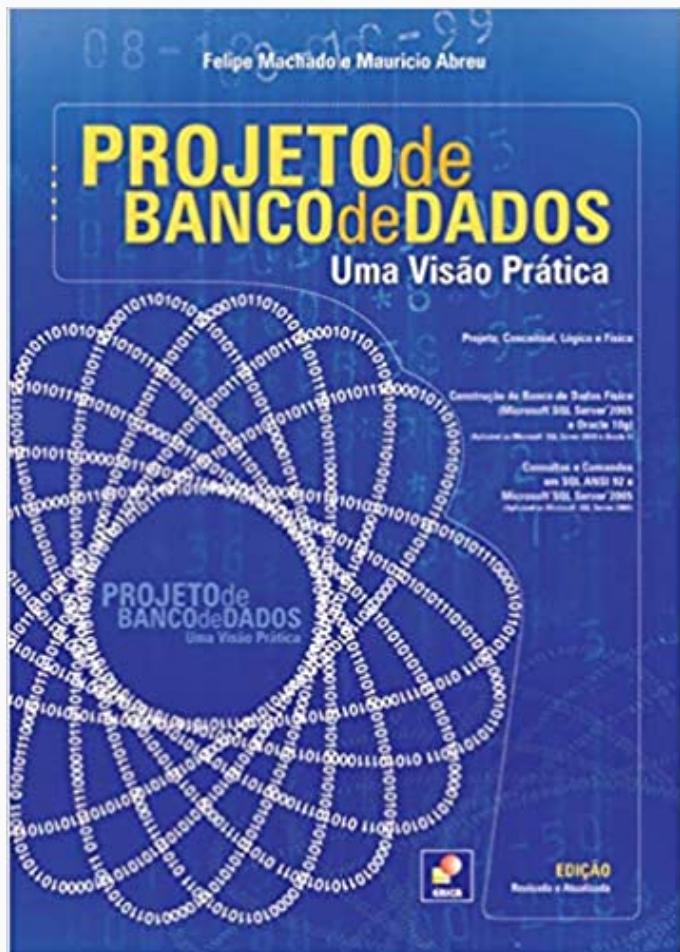
MACHADO, F. N. R.
Banco de Dados: Projeto e Implementação. São Paulo: Érica, 2005.



REFERÊNCIA BIBLIOGÁFICA

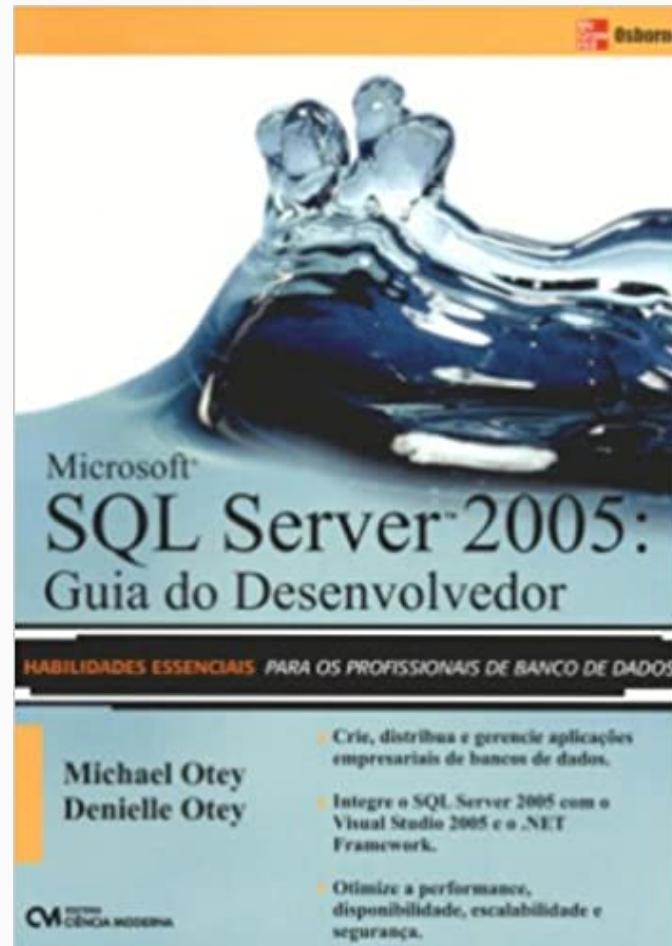


Bibliografia Complementar:



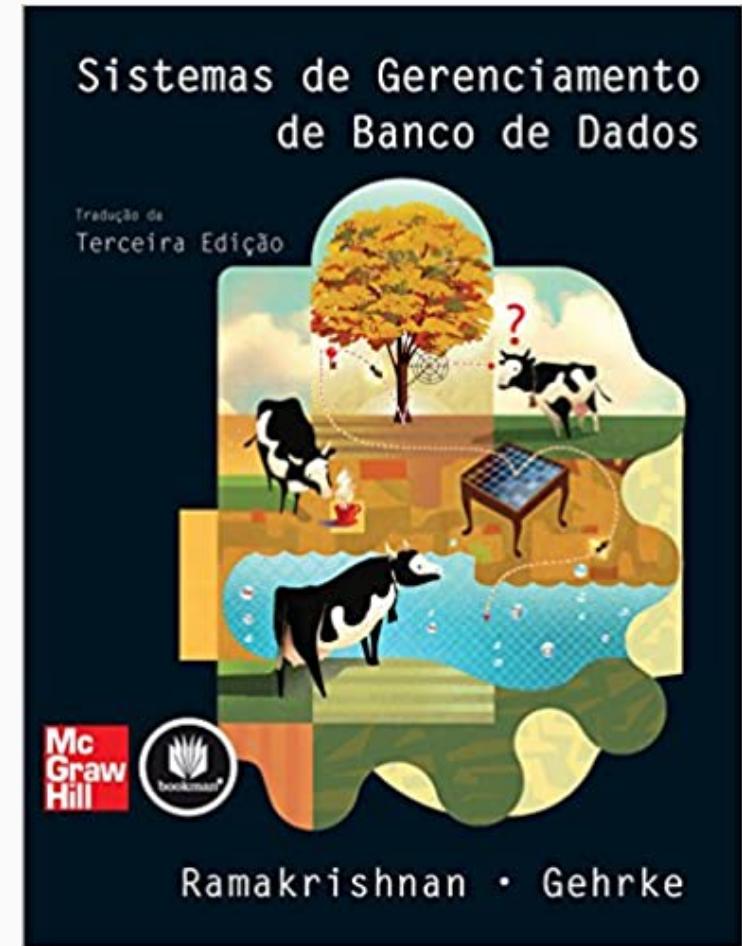
HARRINGTON, J. L.

Projeto de Bancos de Dados Relacionais:
Teoria e Prática. São Paulo: Campus, 2002.



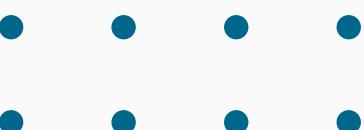
OTEY, M; OTEY, D.

Microsoft SQL Server 2005: Guia do
Desenvolvedor. Rio de Janeiro: Ciência
Moderna, 2007.



RAMAKRISHNAN, R.; GEHRKE, J.

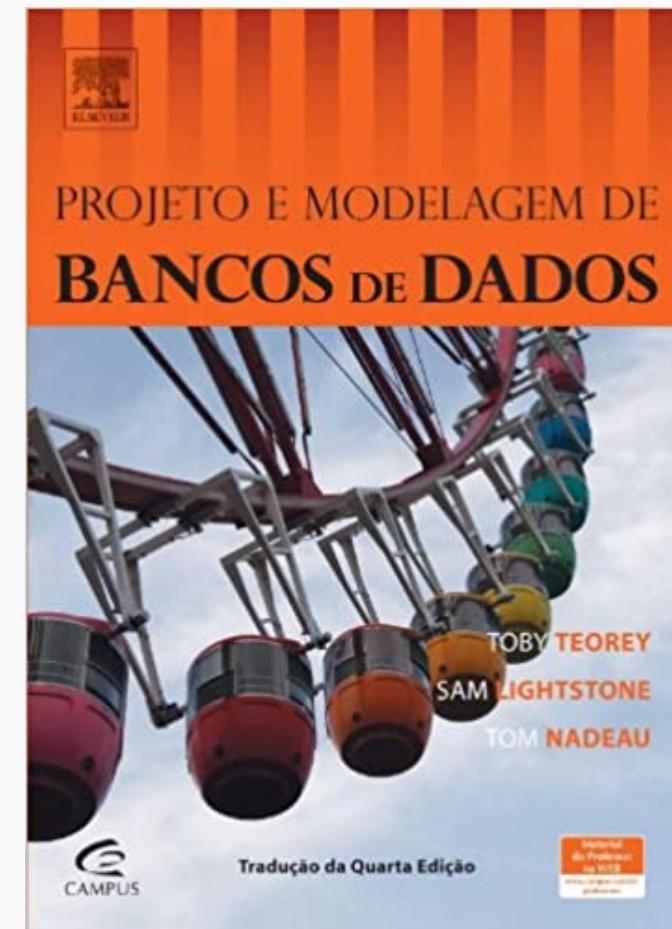
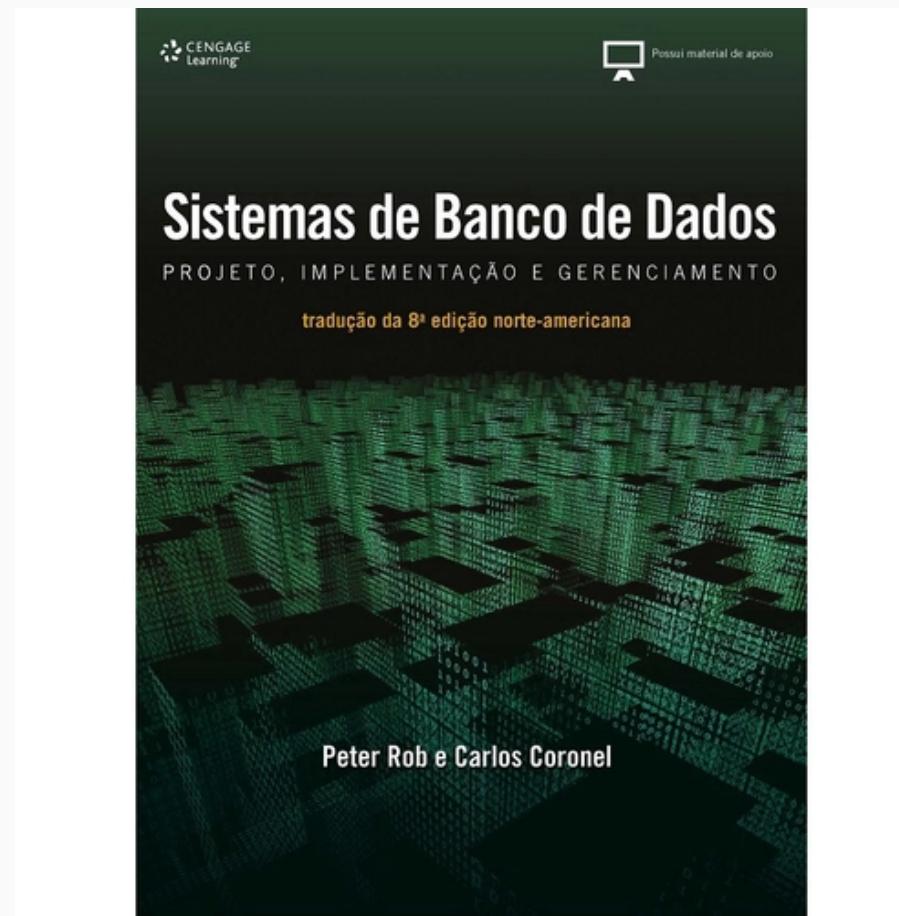
Sistemas de Gerenciamento de Bancos de
Dados. 3 ed. Porto Alegre: Bookman, 2008.



REFERÊNCIA BIBLIOGÁFICA



Bibliografia Complementar:

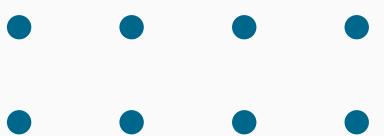


ROB, P; CORONEL, C.

Sistemas de Banco de Dados: Projeto, Implementação e Gerenciamento. 8 ed. São Paulo: Cengage Learning, 2011.

TEOREY, T; LIGHTSTONE, S; NADEAU, T.

Projeto e Modelagem de Bancos de Dados.
São Paulo: Campus, 2006.



REFERÊNCIA BIBLIOGÁFICA



Bibliografia de Referência:



Carvalho, V.

MySQL; Comece com o principal banco de dados open source do mercado. Casa do Código. 2018



Carvalho, V.

PostgreSQL: Banco de dados para aplicações web modernas. Casa do Código. 2017



Gonçalves, E.

SQL: Uma abordagem para bancos de dados Oracle. Casa do Código. 2017



INTRODUÇÃO AO BANCO DE DADOS

O QUE É UM BANCO DE DADOS (BD)



Um banco de dados é uma coleção organizada de informações - ou dados - estruturadas, normalmente armazenadas eletronicamente em um sistema de computador. Um banco de dados é geralmente controlado por um sistema de gerenciamento de banco de dados (DBMS). Juntos, os dados e o DBMS, juntamente com os aplicativos associados a eles, são chamados de sistema de banco de dados, geralmente abreviados para apenas banco de dados.

Os dados nos tipos mais comuns de bancos de dados em operação atualmente são modelados em linhas e colunas em uma série de tabelas para tornar o processamento e a consulta de dados eficientes. Os dados podem ser facilmente acessados, gerenciados, modificados, atualizados, controlados e organizados.



A.C

A.C

Armazenamento Físico



Epopeia de Gilgamesh
Primeiro Livro do Mundo
há 3.200 A.C



Papiro
Primeiro Papel do Mundo
há 2.200 A.C



Biblioteca de Alexandria
Primeiro Acervo do Mundo
há 280 A.C

D.C

D.C

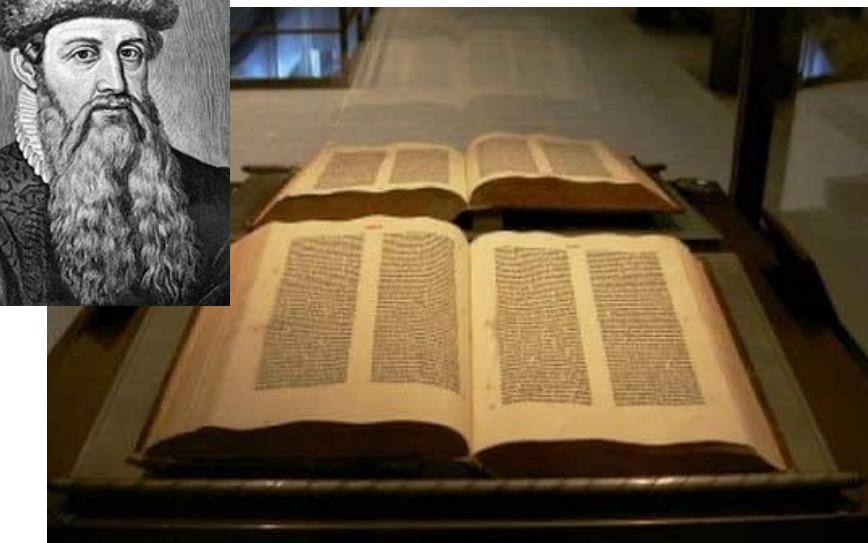
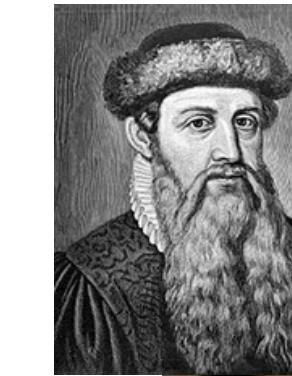
Armazenamento Físico



Pergaminho

**Primeiro Papel para alta produção
há 145 D.C**

**T'sai Lun é conhecido como
inventor do papel**



Biblia Sagrada de Gutenberg

**Primeiro Livro Impresso do Mundo
1455 D.C**

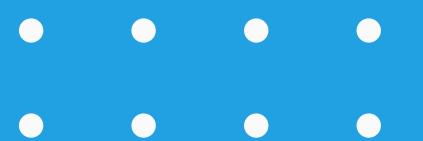
**Gutenberg é conhecido como o Pai ou
Inventor do Papel Impresso**

Armazenamento Físico

- A importância do papel aumentou no final da Idade Média com a expansão do comércio europeu, tornando-se um produto essencial para a **administração pública** e para a **cultura letrada**.
- No fim do século XIX, o crescente aumento da demanda de papel para impressão de livros e jornais e para produtos de consumo levou a criar um novo tipo de papel feito a partir da pasta de madeira (1825).
- Portanto, em razão da evolução industrial foi possível termos o papel que temos hoje



E TUDO ISSO PRODUZIU



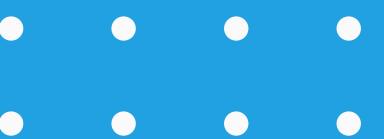


Problemas

- **Perda:** O documento ou arquivo podem ser perdidos durante o manuseio, ou podem ser extraviados, ou danificados (por fogo, água ou recortados).
- **Dificuldade de Acesso:** O documento ou arquivo estão armazenados em locais distantes.
- **Durabilidade:** O documento ou arquivo tem vida útil, assim como seu impresso.
- **Armazenamento:** O documento ou arquivo precisam ser guardados em pastas, arquivos ou pateleiras e nisso, gera um espaço físico a mais no local.
- **Gastos:** A manutenção de climatização, proteção, caixas e mão obras são itens que podem elevar as despesas.
- **Organização:** Necessita de um funcionário para armazenar e indexar os documentos ou arquivos

A PARTIR DESSES PROBLEMAS
COMEÇARAM A PRODUZIR NOVOS
MEIOS DE ARMAZENAMENTO

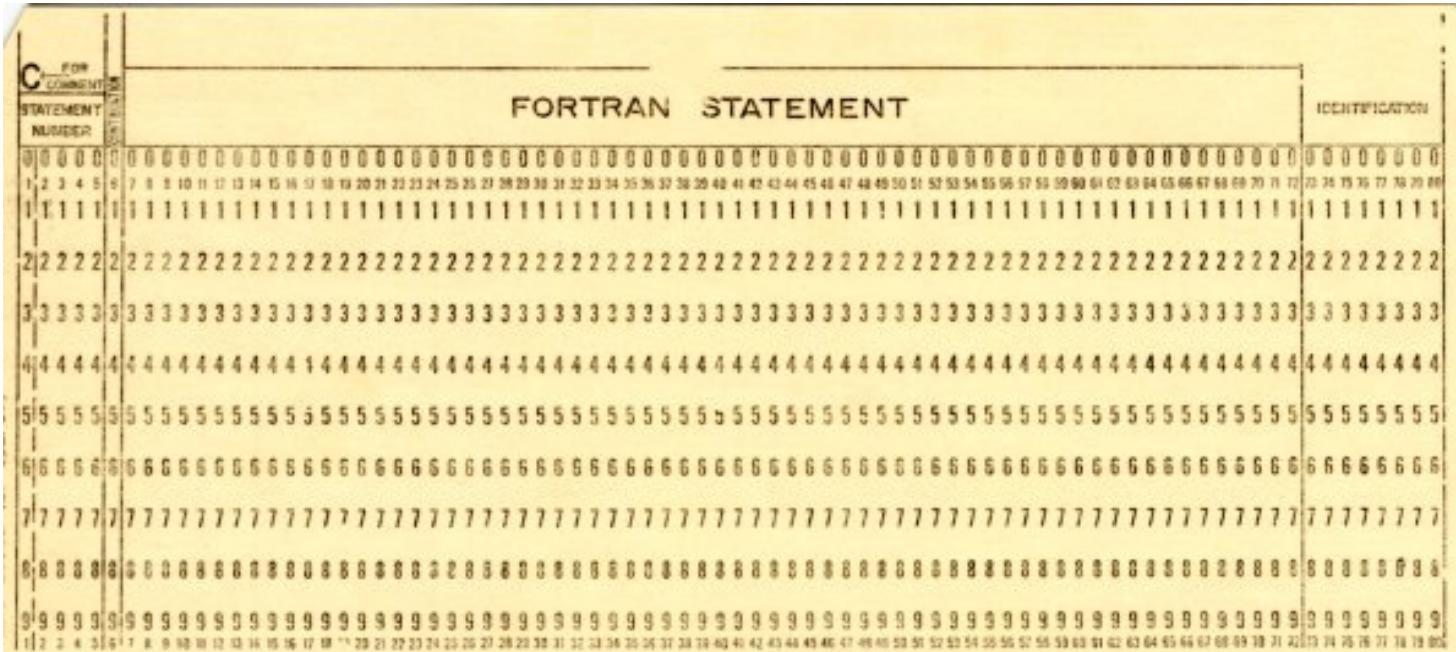


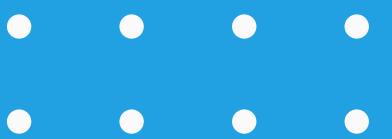


1900

Cartão Perfurado

- Uma das primeiras formas de armazenar dados nesse universo foi com cartões perfurados, criados por Basile Bouchon e aperfeiçoados por Herman Hollerith em 1725.
- Os cartões perfurados armazenam informações simples por meio de buracos estrategicamente posicionados. Quando interpretados por uma máquina, esses furos são decodificados em dados;
- Do ano 1900 a 1950, os cartões perfurados foram o principal meio de entrada de dados, armazenamento de dados e processamento na computação institucional, tudo pela mão da IBM.







1949

Fita Magnética

- Era uma fita plástica coberta com óxido magnético, capaz de armazenar informações;
- As fitas magnéticas, precursoras das fitas cassette, foram responsáveis por uma grande revolução na indústria fonográfica.
- O primeiro computador a utilizar as fitas, o UNIVAC I



UNIVAC I

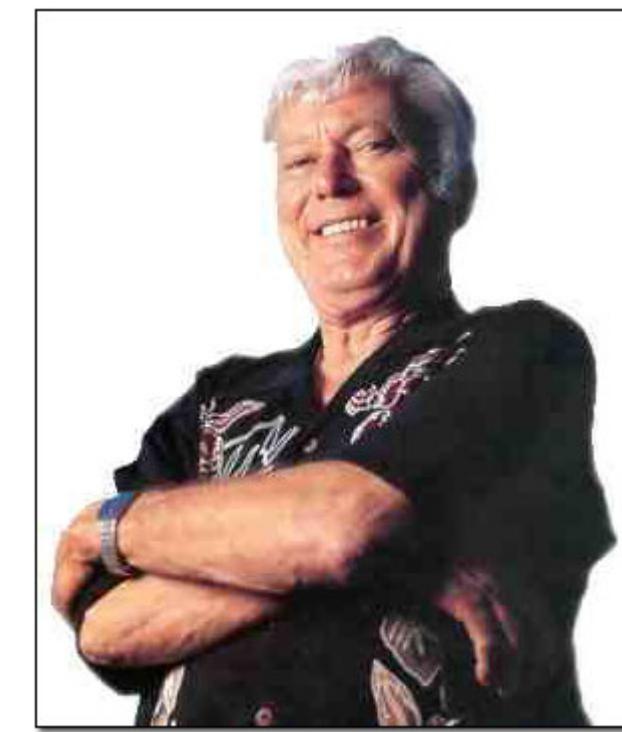
1953

Primeiro Disco Rígido

- A IBM embarca o primeiro disco rígido no sistema RAMAC 305. O tamanho total é de 5MB a um custo de \$ 10.000 um megabyte. O sistema é tão grande quanto dois refrigeradores e usa 50 pratos de 24 polegadas de diâmetro;
- Alan Shugart, engenheiro da IBM na época e que viria a fundar a Seagate disse: "Começaram do 0 no laboratório, o RAMAC era um sistema completo e não somente um disco rígido. Ninguém havia feito um disco rígido até então



RAMAC 305



Alan Shugart
(1930-2006)

1953

Primeiro Disco Rígido

- Segundo Shugart a ideia principal do disco rígido era o acesso aleatório (Random Access)
- Para poder ler e escrever no disco as cabeças de leitura deveriam se movimentar pelas trilhas e, segundo Shugart, a melhor saída foi uma pilha de discos
- O sistema era capaz de atingir 1200 rpm
- A tinta que foi utilizada nos discos era similar à tinta que pintou a ponte Golden Gate
- Foram necessários 12 protótipos



RAMAC 305

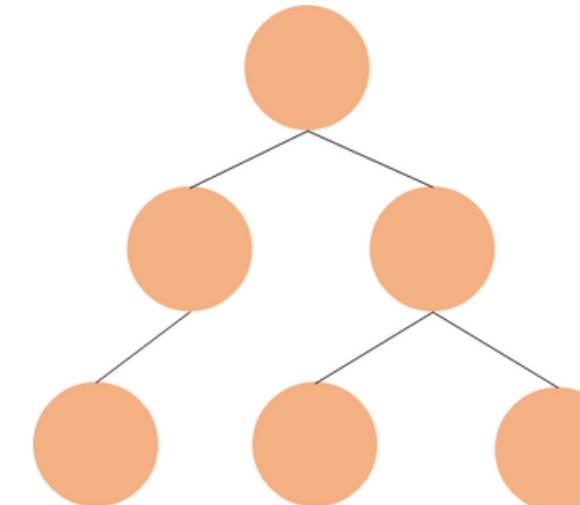


1960

Os Primeiros Modelos de Dados

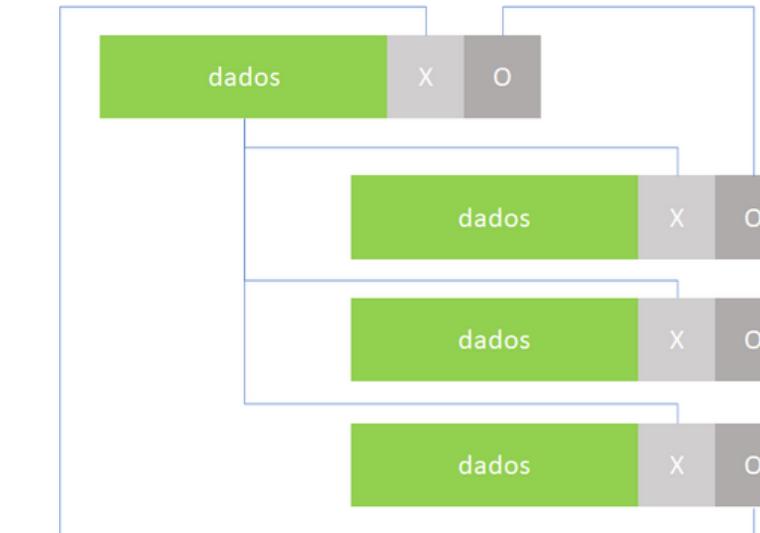
- Os fundamentos de bancos de dados surgiram na empresa IBM na década de 1960 através de pesquisas de funções de automação de escritório
- As empresas descobriram que estava muito custoso雇用 um número grande de pessoas para fazer trabalhos como armazenar e indexar (organizar) arquivos
- Diante disso, surgiram dois modelos:

**MODELO HIERÁRQUICO
(IMS - INFORMATION MANAGEMENT SYSTEM).**

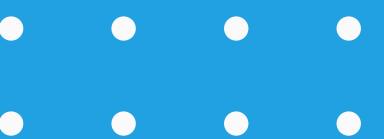


- Também se utilizava de registros para representar os dados e links para os relacionamentos;
- São organizados na forma de uma árvore com raiz;

**MODELO EM REDE
(CODASYL - COMMITTEE FOR DATA SYSTEMS LANGUAGE)**



- Os primeiros trabalhos foram realizados em 1964 por Charles Bachman;
- Dados são representados por uma coleção de registros e os relacionamentos por meio de links;
- É representado por um diagrama constituído por caixas e linhas;

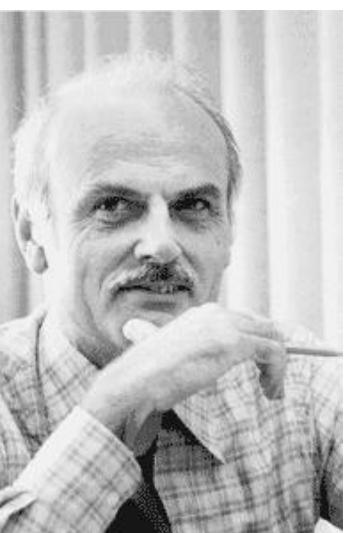


1970

Modelo Relacional

- Estes modelo se baseia no princípio de que todos os dados estão guardados em tabelas e, que podem ser relacionados com outras tabelas.
- Esse conceito foi criado por **Edgar Frank Codd**, e descrito no artigo "*Relational Model of Data for Large Shared Data Banks*", sendo o primeiro modelo de dados descrito teoricamente.
- O modelo relacional se tornou um sucessor do modelo hierárquico e de rede e, permitiu que inúmeros trabalhos ou modelos fossem baseados a partir dele.
- Esse modelo tem como características:
 - Escalonamento Vertical;
 - Esquema Predefinido (Tabelas e Colunas);
 - Prioriza a Consistência e a Qualidade do Dado;
 - Tabelas se relacionam com outras Tabelas;

DR. EDGAR FRANK CODD (1923 - 2003)
O PAI DO MODELO RELACIONAL



Information Retrieval

P. BAXENDALE, Editor

A Relational Model of Data for Large Shared Data Banks

E. F. CODD

IBM Research Laboratory, San Jose, California

Future users of large data banks must be protected from having to know how the data is organized in the machine (the internal representation). A prompting service which supplies such information is not a satisfactory solution. Activities of users at terminals and most application programs should remain unaffected when the internal representation of data is changed and even when some aspects of the external representation are changed. Changes in data representation will often be needed as a result of changes in query, update, and report traffic and natural growth in the types of stored information.

Existing noninferential, formatted data systems provide users with tree-structured files or slightly more general network models of the data. In Section 1, inadequacies of these models are discussed. A model based on *n*-ary relations, a normal form for data base relations, and the concept of a universal data sublanguage are introduced. In Section 2, certain operations on relations (other than logical inference) are discussed and applied to the problems of redundancy and consistency in the user's model.

KEY WORDS AND PHRASES: data bank, data base, data structure, data organization, hierarchies of data, networks of data, relations, derivability, redundancy, consistency, composition, join, retrieval language, predicate calculus, security, data integrity

CR CATEGORIES: 3.70, 3.73, 3.75, 4.20, 4.22, 4.29

1. Relational Model and Normal Form

1.1. INTRODUCTION

This paper is concerned with the application of elementary relation theory to systems which provide shared access to large banks of formatted data. Except for a paper by Childs [1], the principal application of relations to data systems has been to deductive question-answering systems. Leveine and Maron [2] provide numerous references to work in this area.

In contrast, the problems treated here are those of *data independence*—the independence of application programs and terminal activities from growth in data types and changes in data representation—and certain kinds of *data inconsistency* which are expected to become troublesome even in nondeductive systems.

The relational view (or model) of data described in Section 1 appears to be superior in several respects to the graph or network model [3, 4] presently in vogue for non-inferential systems. It provides a means of describing data with its natural structure only—that is, without superimposing any additional structure for machine representation purposes. Accordingly, it provides a basis for a high level data language which will yield maximal independence between programs on the one hand and machine representation and organization of data on the other.

A further advantage of the relational view is that it forms a sound basis for treating derivability, redundancy, and consistency of relations—these are discussed in Section 2. The network model, on the other hand, has spawned a number of confusions, not the least of which is mistaking the derivation of connections for the derivation of relations (see remarks in Section 2 on the “connection trap”).

Finally, the relational view permits a clearer evaluation of the scope and logical limitations of present formatted data systems, and also the relative merits (from a logical standpoint) of competing representations of data within a single system. Examples of this clearer perspective are cited in various parts of this paper. Implementations of systems to support the relational model are not discussed.

1.2. DATA DEPENDENCIES IN PRESENT SYSTEMS

The provision of data description tables in recently developed information systems represents a major advance toward the goal of data independence [5, 6, 7]. Such tables facilitate changing certain characteristics of the data representation stored in a data bank. However, the variety of data representation characteristics which can be changed without logically impairing some application programs is still quite limited. Further, the model of data with which users interact is still cluttered with representational properties, particularly in regard to the representation of collections of data (as opposed to individual items). Three of the principal kinds of data dependencies which still need to be removed are: ordering dependence, indexing dependence, and access path dependence. In some systems these dependencies are not clearly separable from one another.

1.2.1. Ordering Dependence. Elements of data in a data bank may be stored in a variety of ways, some involving no concern for ordering, some permitting each element to participate in one ordering only, others permitting each element to participate in several orderings. Let us consider those existing systems which either require or permit data elements to be stored in at least one total ordering which is closely associated with the hardware-determined ordering of addresses. For example, the records of a file concerning parts might be stored in ascending order by part serial number. Such systems normally permit application programs to assume that the order of presentation of records from such a file is identical to (or is a subordering of) the

1976

Modelo Entidade-Relacionamento (MER)

- **Peter Chen**, desenvolveu o modelo capaz de **descrever** um modelo de dados e ilustrar o aspectos de informação de um domínio de negócios ou seus requisitos.
- Denominado de Modelo Entidade-Relacionamento (MER), o mesmo é utilizado para representar uma implementação de um banco de dados relacional.
- Os principais componentes desse modelo são as **Entidades, os Relacionamentos e as Cardinalidades**.



DR. PETER CHEN
CRIADOR DO MODELO ER.

The Entity-Relationship Model—Toward a Unified View of Data

PETER PIN-SHAN CHEN
Massachusetts Institute of Technology

A data model, called the entity-relationship model, is proposed. This model incorporates some of the important semantic information about the real world. A special diagrammatic technique is introduced as a tool for database design. An example of database design and description using the model and the diagrammatic technique is given. Some implications for data integrity, information retrieval, and data manipulation are discussed.

The entity-relationship model can be used as a basis for unification of different views of data: the network model, the relational model, and the entity set model. Semantic ambiguities in these models are analyzed. Possible ways to derive their views of data from the entity-relationship model are presented.

Key Words and Phrases: database design, logical view of data, semantics of data, data models, entity-relationship model, relational model, Data Base Task Group, network model, entity set model, data definition and manipulation, data integrity and consistency
CR Categories: 3.50, 3.70, 4.33, 4.34

1. INTRODUCTION

The logical view of data has been an important issue in recent years. Three major data models have been proposed: the network model [2, 3, 7], the relational model [8], and the entity set model [25]. These models have their own strengths and weaknesses. The network model provides a more natural view of data by separating entities and relationships (to a certain extent), but its capability to achieve data independence has been challenged [8]. The relational model is based on relational theory and can achieve a high degree of data independence, but it may lose some important semantic information about the real world [12, 15, 23]. The entity set model, which is based on set theory, also achieves a high degree of data independence, but its viewing of values such as "3" or "red" may not be natural to some people [25].

This paper presents the entity-relationship model, which has most of the advantages of the above three models. The entity-relationship model adopts the more natural view that the real world consists of entities and relationships. It

Copyright © 1976, Association for Computing Machinery, Inc. General permission to republish, but not for profit, all or part of this material is granted provided that ACM's copyright notice is given and that reference is made to the publication, to its date of issue, and to the fact that reprinting privileges were granted by permission of the Association for Computing Machinery. A version of this paper was presented at the International Conference on Very Large Data Bases, Framingham, Mass., Sept. 22-24, 1975.

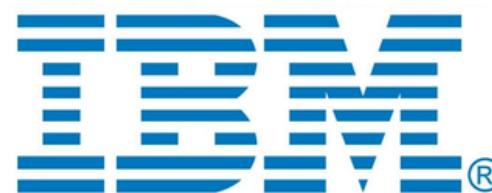
Author's address: Center for Information System Research, Alfred P. Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA 02139.

ACM Transactions on Database Systems, Vol. 1, No. 1, March 1976, Pages 9-36.

1974

IBM System R

- Desenvolvido pela IBM San Jose e serviu de base para o IBM SQL/DS, IBM DB2, Oracle e todos os SGBDS existentes;
- Foi o Primeiro Banco de Dados a Utilizar SQL, entretanto a IBM chamava de **SEQUEL**



**COMPUTING
PRACTICES**

A History and Evaluation of System R

Donald D. Chamberlin Thomas G. Price
Morton M. Astrahan Franco Putzolu
Michael W. Blasgen Patricia Griffiths Selinger
James N. Gray Mario Schkolnick
W. Frank King Donald R. Slutz
Bruce G. Lindsay Irving L. Traiger
Raymond Lorie Bradford W. Wade
James W. Mehl Robert A. Yost

IBM Research Laboratory
San Jose, California

1. Introduction

Throughout the history of information storage in computers, one of the most readily observable trends has been the focus on data independence. C.J. Date [27] defined data independence as "immunity of applications to change in storage structure and access strategy." Modern database systems offer data independence by providing a high-level user interface through which users deal with the information content of their data, rather than the various bits, pointers, arrays, lists, etc. which are used to represent that information. The system assumes responsibility for choosing an appropriate internal

SUMMARY: System R, an experimental database system, was constructed to demonstrate that the usability advantages of the relational data model can be realized in a system with the complete function and high performance required for everyday production use. This paper describes the three principal phases of the System R project and discusses some of the lessons learned from System R about the design of relational systems and database systems in general.

representation for the information; indeed, the representation of a given fact may change over time without users being aware of the change.

The relational data model was proposed by E.F. Codd [22] in 1970 as the next logical step in the trend toward data independence. Codd observed that conventional database systems store information in two ways: (1) by the contents of records stored in the database, and (2) by the ways in which these records are connected together. Different systems use various names for the connections among records, such as links, sets, chains, parents, etc. For example, in Figure 1(a), the fact that supplier Acme supplies bolts is repre-

sented by connections between the relevant part and supplier records. In such a system, a user frames a question, such as "What is the lowest price for bolts?", by writing a program which "navigates" through the maze of connections until it arrives at the answer to the question. The user of a "navigational" system has the burden (or opportunity) to specify exactly how the query is to be processed; the user's algorithm is then embodied in a program which is dependent on the data structure that existed at the time the program was written.

Relational database systems, as proposed by Codd, have two important properties: (1) all information is

632

Communications

of

the ACM

October 1981
Volume 24
Number 10

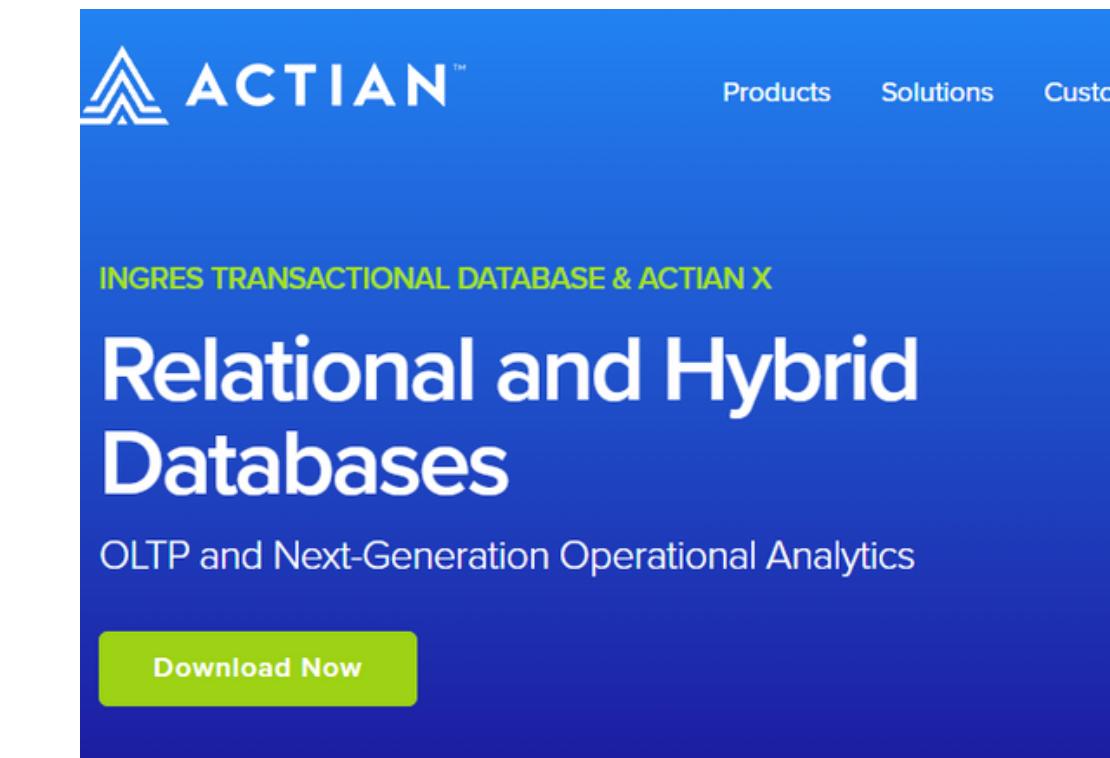
Um documento que conta a história e avaliação do System R

1974

INGRES

- Desenvolvido pelos professores Michael Stonebraker e Eugene Wong UCB.
- O INGRES foi um concorrente direto do System R da IBM.
- O mesmo só foi comercializado a partir da década de 80
- Serviu como base para Sybase, Postgres o MS SQL Server.
- Este sistema utilizava **QUEL** como linguagem de consulta.
- Atualmente o INGRES pertence ACTIAN e se encontra na versão 11.0 (desde 2017)

INGRES



Disponível em:

<https://www.actian.com/data-management/actian-x-hybrid-rdbms/>



1978 ORACLE v2

- É o Primeiro Sistema Gerenciador de Banco de Dados da História
- Desenvolvido pela ORACLE CORPORATION;
- Sua primeira versão foi lançada em 1978 (44 anos) - **É considerado o SGBDR com maior tempo de atuação**



1979 - 1983



Imagen do Oracle V4

Disponível em:
<https://www.oracle.com/>



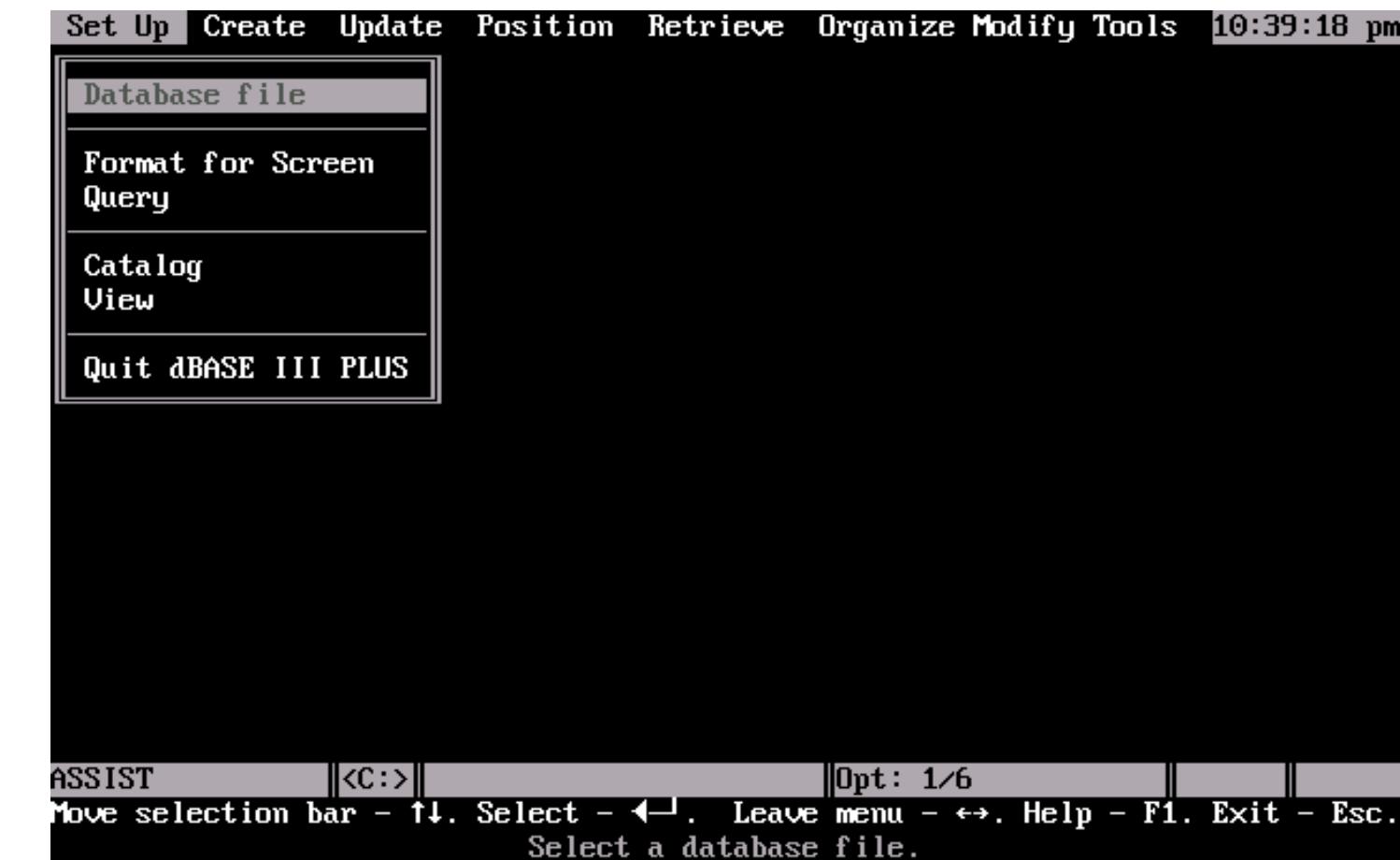


1979 dBASE I

- É um Banco de dados que implementa nativamente (embutido)
- Desenvolvido pela Wayne Ratliff;
- É considerado o segundo SGBD com maior tempo de atuação (43 anos).

dBaseTM

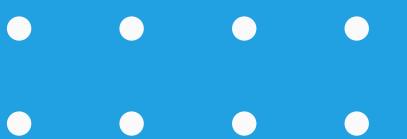
Disponível em:
<https://www.dbase.com/>



70-79

Alguns Pontos Chave sobre a DEC. 70

- A criação do modelo relacional
- A documentação em banco de dados
- Surgimento dos primeiros Bancos de Dados voltados à Indústria
- Criação do QUEL e SEQUEL, que serviram como base ao SQL

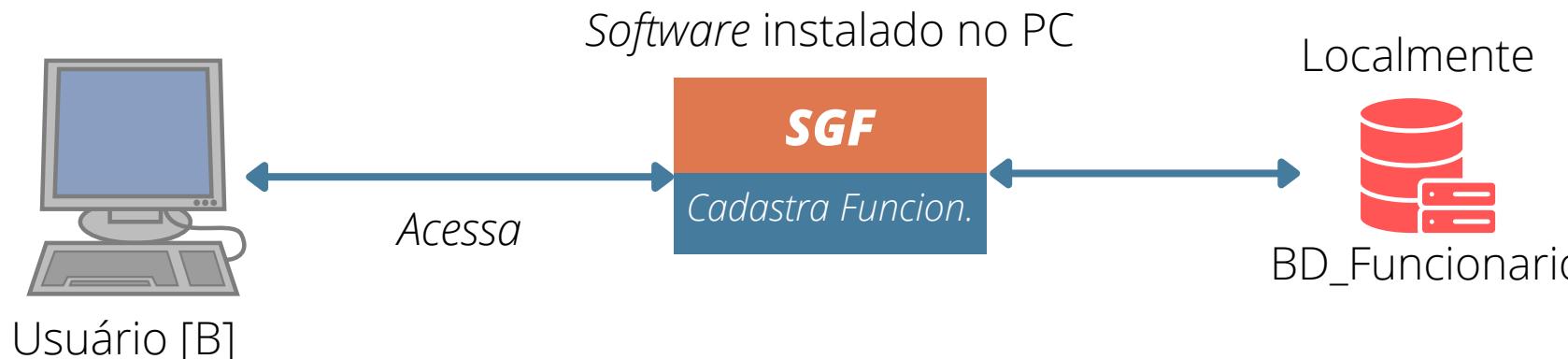
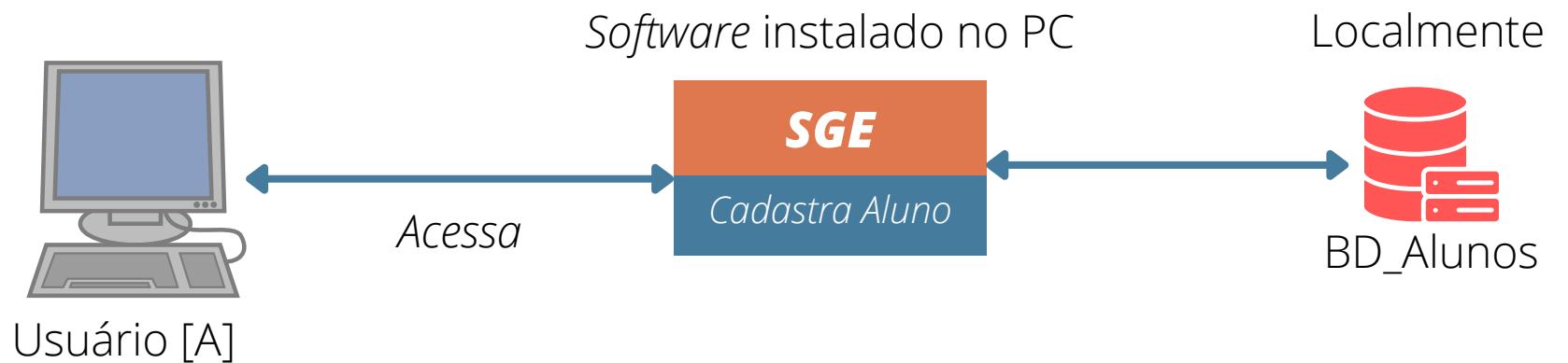


1980

SISTEMA GERENCIADOR DE BANCO DE DADOS RELACIONAL (SGBDR)

- *Relational Data Base Management System* (RDBMS) ou Sistema Gerenciador de Banco de Dados Relacional (SGBDR) tem o princípio em Gerenciar N Banco de Dados dentro uma infraestrutura/negócio.
- O SGBDR tem como características:
 - Manipulação de dados;
 - Definição de estruturas de dados (Tabelas e Colunas);
 - Segurança contra falhas e acessos não autorizados;
 - Controle de uso compartilhado dos dados por diversos usuários;
 - Manter a integridade dos dados (Atender as características do ACID)

Antes do SGBDR



Problemas

- Equipamento exclusivo para execução do *software* e armazenamento de dados;
- Apenas uma pessoa de cada vez poderia acessar;
- Se o equipamento danificasse, a base de dados era comprometida;
- Dados Isolados;

1985

TCP/IP

- São dois protocolos que garantem a entrega de pacotes ao seu destino de forma correta e segura;
- Em outras palavras o TCP (Protocolo de Controle de Transmissão) e IP (Protocolo de Internet) visam a interconexão dos inúmeros dispositivos na rede, isto é, especificam como a comunicação dever ser realizadas;
- Sua importância é enorme. Sem a presença desse conjunto, seria impossível definir os padrões de transmissão, que são adotados por diferentes dispositivos das mais variadas aplicações.
- No cenário Banco de dados a importância é a mesma, uma vez que as máquinas-cliente necessitam conectar-se ao servidor

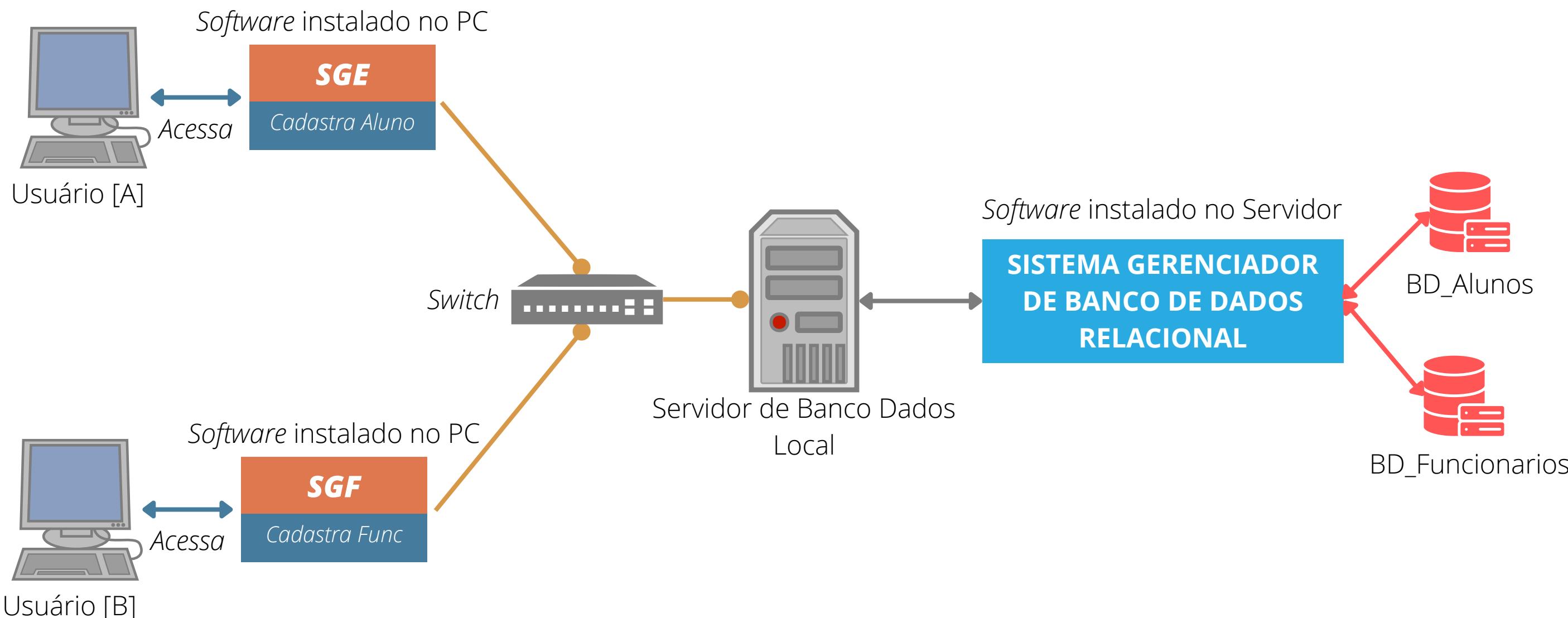
1986

Linguagem Estruturada de Consulta (SQL)

- **Structured Query Language (SQL)** ou Linguagem Estruturada de Consulta se torna um padrão mundial.
- Os modelos em rede e hierárquico passam a ficar em segundo plano praticamente sem desenvolvimentos utilizando seus conceitos, porém vários sistemas legados continuam em uso;
- Embora o SQL tenha sido originalmente criado pela IBM, houve-se a necessidade de ser criado e adaptado um padrão para a linguagem.
- Esta tarefa foi realizada pela *American National Standards Institute (ANSI)* em 1986 e *International Organization for Standardization (ISO)* em 1987. Portanto, para cada versão lançada é se utilizado sql[ano]. exemplo: sql-92 (de 1992), sql-3 (de 2003), sql-16 (de 2016)



SGBDR antes da Internet Comercial



Outro conceito que pode ser atribuído a esse diagrama é a **Intranet**, uma rede interna, corporativa e que não permite acesso externo

1988

ORACLE v6

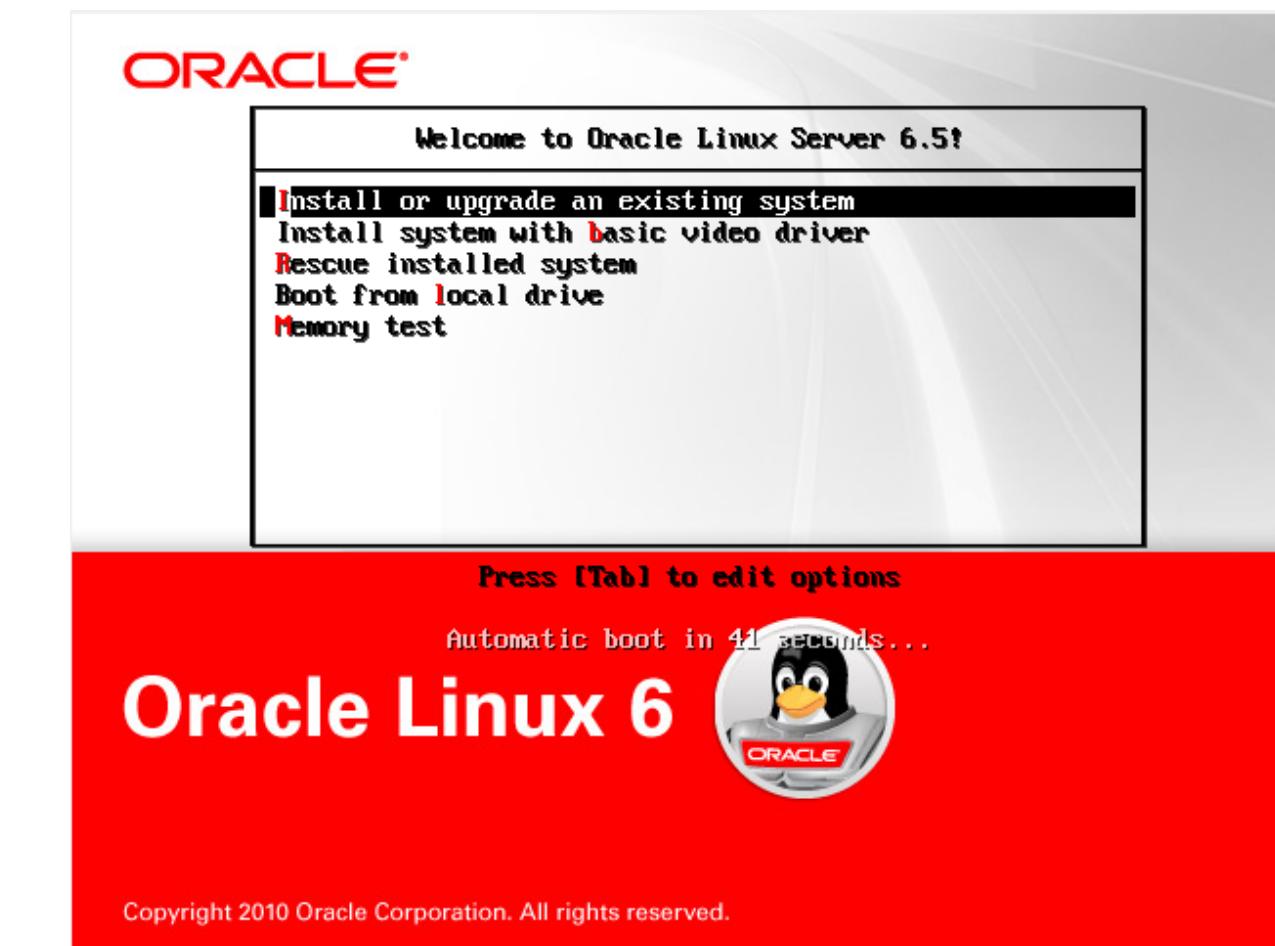
- É o Primeiro Sistema Gerenciador de Banco de Dados da História a utilizar oficialmente SQL definido pela ANSI e ISO;
- Foi a primera versão a oferecer suporte escalabilidade, backup e recuperação online
- Foi a primera versão a oferecer PL/SQL (Procedures, Triggers, Functions em SQL)
- Atualmente se encontra na versão ORACLE 19c (2019)



1983 - 1995



Disponível em:
<https://www.oracle.com/>



80-89

Alguns Pontos Chave sobre a DEC. 80

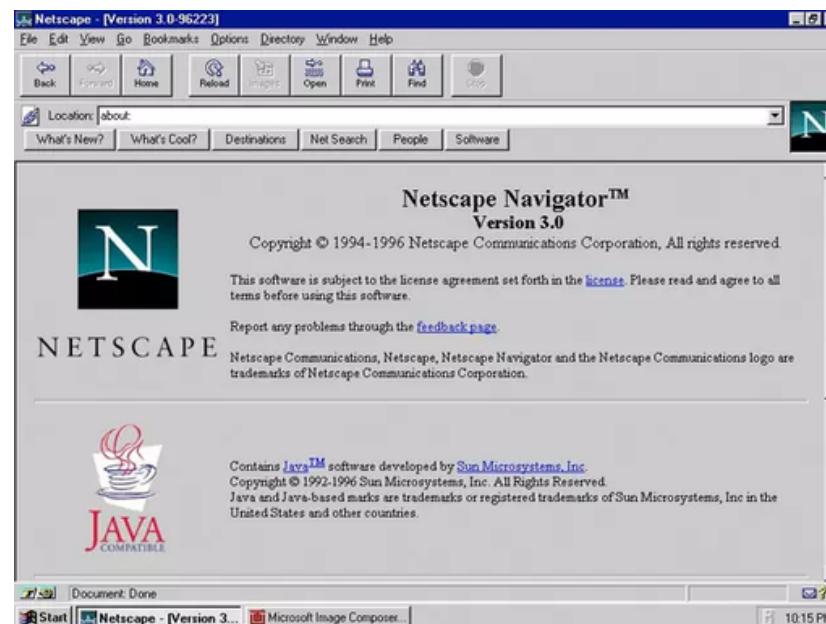
- O Surgimento dos primeiros SGBDs
- A criação do SQL pela IBM
- A Padronização do SQL pela ANSI e a ISO (SQL-86 e SQL-89)
- A evolução da computação na industria e a necessidade de armazenar informação centralizada



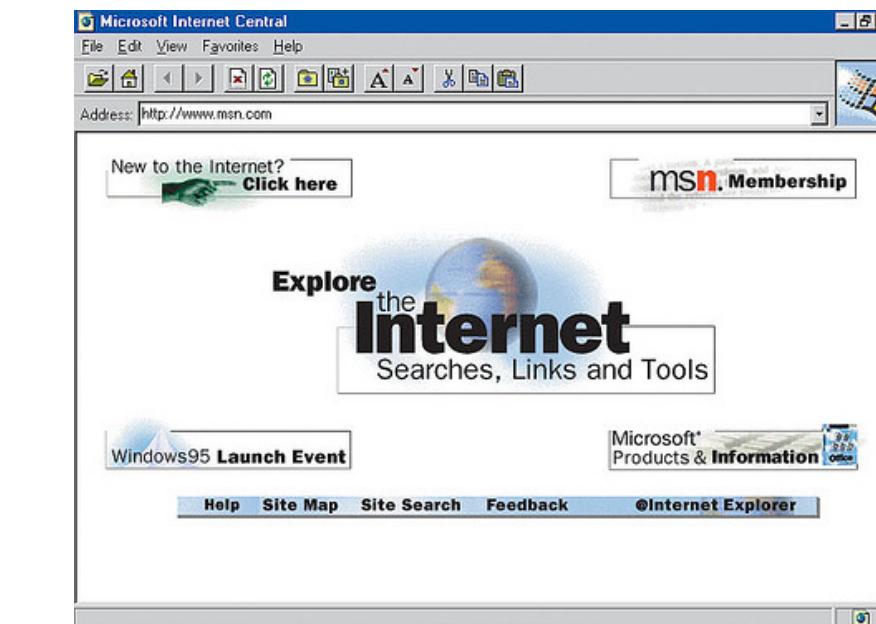
1990

Crescimento da Internet

- Através do protocolo **World Wide Web (WWW)** permitiu a distribuição de documentos hipertexto (HTTP) acessíveis ao navegador web à Internet.
- Diante desse novo cenário, começou a surgir novos nichos/categorias de comércio. Denominado de **Comércio Eletrônico ou Digital**, onde usuários poderiam comprar e usufruir produtos à distância;
- Em razão dessa alta demanda, houve-se a necessidade de aperfeiçoar os Servidores de Banco de Dados, assim como o surgimento de novos softwares para SGBDR, que permiti-se o funcionamento de uma aplicação WEB por 24x7 (24h por dia, 7 dias da semana)



Netscape (Navegador popular na época)



Internet Explorer da Microsoft (Navegador popular na época)

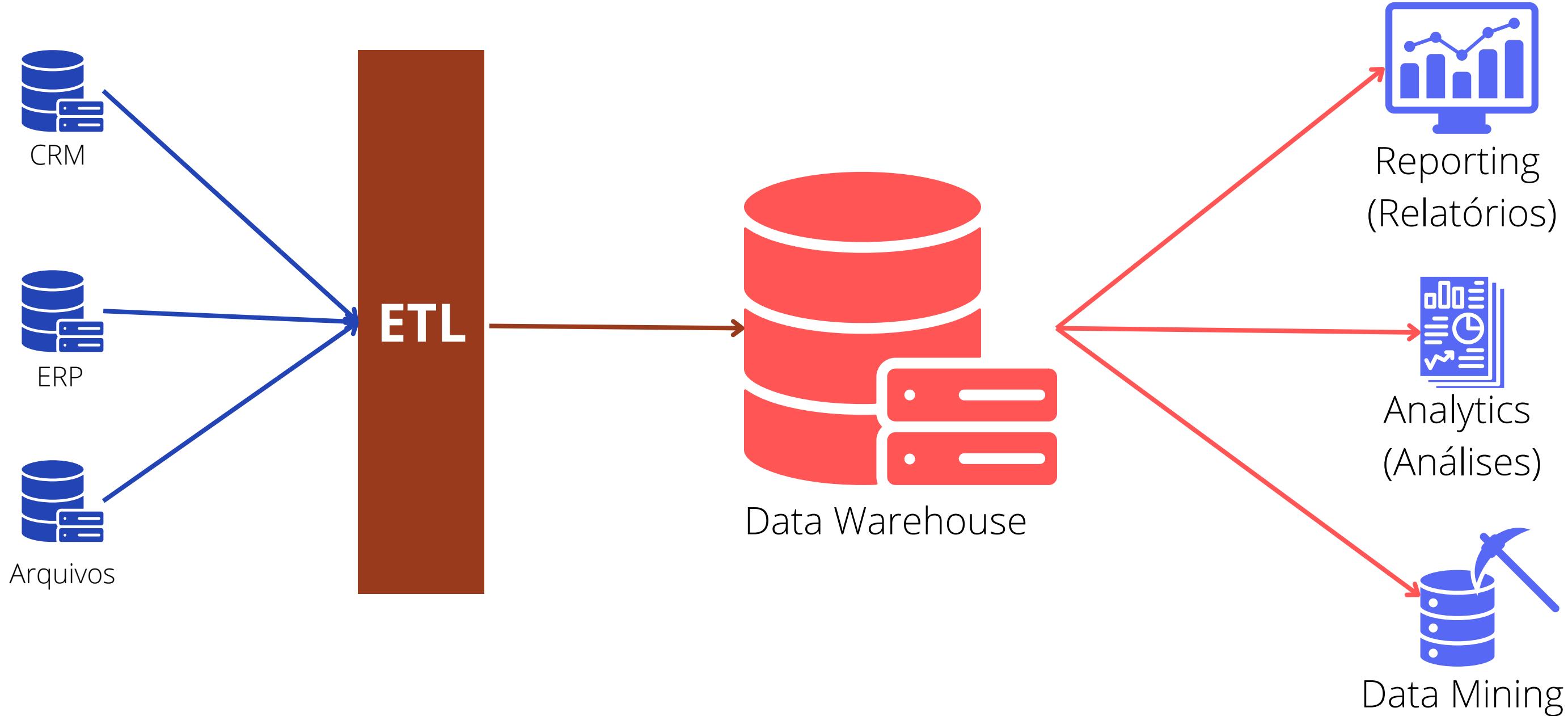
1990

Data Warehouse (DW)

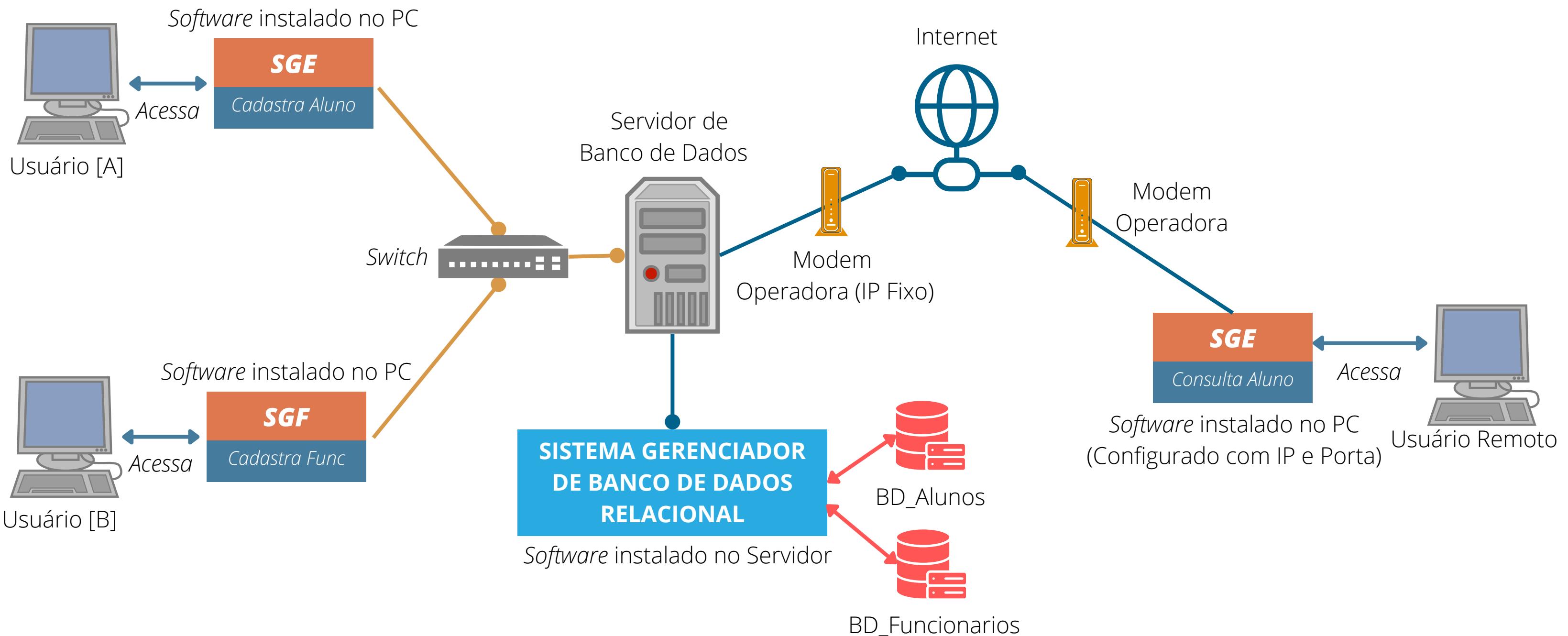
- **Data Warehouse (DW)** ou Armazém de dados, é utilizado para armazenar informações relativas e significantes sobre um negócio (ou empresa), obtidos por inúmeros bancos de dados.
- O **DW** possibilita armazenar grande volumes de dados para favorecer a criação de relatórios e análises estratégicas, para possibilitar ou facilitar uma tomada de decisão.
- A partir do **DW** surgiu outras ferramentas/conceitos:
 - **OLTP** (Online Transaction Processing): são sistemas que têm a tarefa de monitorar e processar as funções básicas e rotineiras de uma organização (faturamento, estoque e etc.);
 - **OLAP** (Online Analytical Processing): são sistemas que têm a tarefa em navegar pelos os dados de um DW;
 - **ETL** (Extract Transform Load): são sistemas que têm a tarefa em extrair dados em diversos bancos de dados e carrega-los em um DW.
 - **Data mining**: Processo de mineção de dados ou descoberta de padrões existentes sob um DW;
 - **BI**: É o resultado de um processo/análise de dados que pode gerar uma conhecimento a uma determinada empresa/pessoa.

1990

Data Warehouse (DW)



SGBDR depois da Internet Comercial



Esse diagrama é baseado sobre os softwares em desktop (com ou sem interface gráfica) que eram muito populares na década de 90

1990

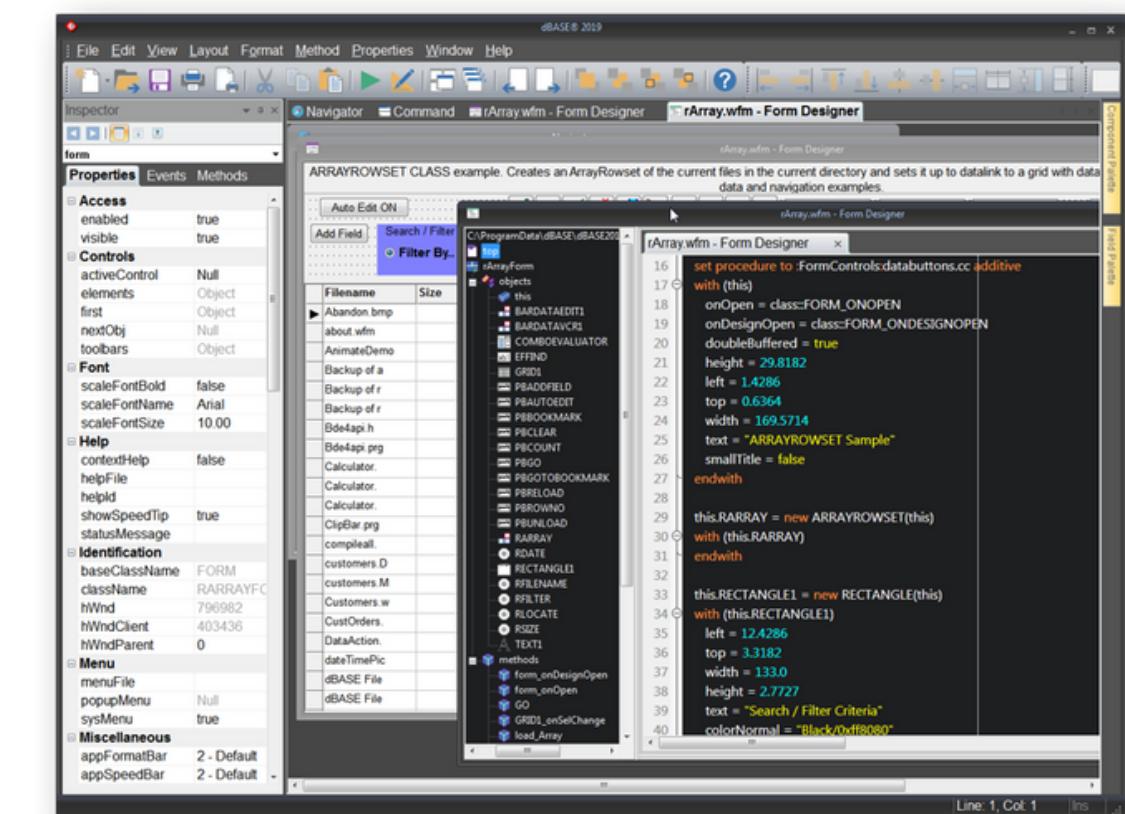
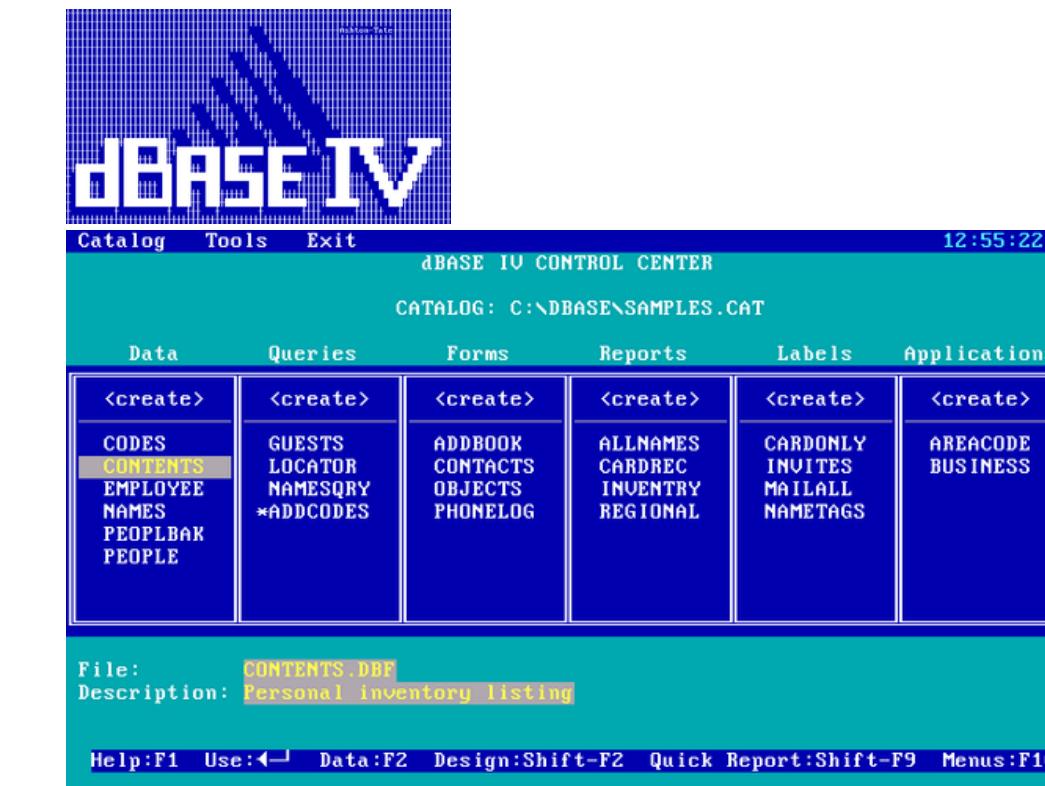
dBASE IV

- É um SGBDR que implementa um Banco de dados SQL nativamente (embutido)
- Desenvolvido pela Wayne Ratliff;
- Foi amplamente utilizado pela Apple e a Microsoft
- Foi a primeira versão a incluir o SQL como script principal
- A ultima versão é dBASE 2019



1990 - Atualmente

Disponível em:
<https://www.dbase.com/>



1990

Microsoft SQL Server

- SGDBR;
- Desenvolvido pela Microsoft e Sybase (inicialmente);
- Multiplataforma (Linux, Windows e etc);
- Sua primeira versão foi lançada em 1989 no OS/2 e Apenas em 1993 no Windows
- A cada 2 anos é lançado uma versão do SQL Server. No momento se encontra na versão 16.0, sendo a 15.0 como a versão estável e 10.25 como a versão mínima



1990 - 2004



2005 - 2008



2009 - Atualmente

Disponível em:

<https://www.microsoft.com/pt-br/sql-server/sql-server-2019>

1993

IBM Db2

- SGDBR;
- Lançado em 1983, mas apenas em na versão 3.4 foi possível utilizar SQL
- Desenvolvido pela IBM;
- Foi o Sucessor do *System R*
- Atualmente o Db2 se encontrar na versão 11.5 e possui um grande suporte no serviço de nível da IBM



Disponível em:

<https://www.ibm.com/products/db2>

A screenshot of the IBM DB2 Control Center interface. The title bar reads "DB2 Control Center - localhost - Signed in as db2admin Default user id = db2admin". The menu bar includes Control Center, Selected, Edit, View, Tools, and Help. The toolbar has various icons for database management tasks. The left pane shows a tree view of the database system structure under "Systems" (MIKEWIKI), "Instances" (DB2), and "Databases" (SAMPLE). The right pane displays a table titled "MIKEWIKI - DB2 - SAMPLE - TABLES" with columns: Name, Schema, Type, and Table space. The table lists numerous system tables, all belonging to the "db2admin" schema and the "USERSPACE1" table space. The table includes rows like AAA, SYSATTRIB_, SYSPUFFE_, SYSPUFFE_, SYSCHECKS, SYSDOLAU_, SYSDOLC_, SYSDOLDL_, SYSDOLOP_, SYSDOLPR_, SYSDOLU_, SYSCONST_, SYSDATAT_, SYSDIGAUTH, and SYSEDEFNL.



1995 MySQL

- SGDBR;
- Desenvolvido pela Oracle;
- Multiplataforma (Linux, macOS, Windows e etc);
- Sua primeira versão foi lançada em 23 de maio de 1995 (27 anos)
- Atualmente se encontra na versão 8.0 com suporte a 28 idiomas



1995 - Atualmente

The screenshot shows the MySQL Workbench application. In the Navigator pane, there are three schemas listed: sakila, sys, and world. The SQL Editor pane contains the command "SHOW ENGINES". The Result Grid pane displays a table of storage engines with the following data:

Engine	Support	Comment	Transactions	XA	Savepoints
MEMORY	YES	Hash based, stored in memory, useful for temp...	NO	NO	NO
MRG_MYISAM	YES	Collection of identical MyISAM tables	NO	NO	NO
CSV	YES	CSV storage engine	NO	NO	NO
FEDERATED	NO	Federated MySQL storage engine	N/A	N/A	N/A
PERFORMANCE_SCHEMA	YES	Performance Schema	NO	NO	NO
MyISAM	YES	MyISAM storage engine	NO	NO	NO
InnoDB	DEFAULT	Supports transactions, row-level locking, and fo...	YES	YES	YES
BLACKHOLE	YES	/dev/null storage engine (anything you write to ...	NO	NO	NO
ARCHIVE	YES	Archive storage engine	NO	NO	NO

Interface do Workbench do MySQL



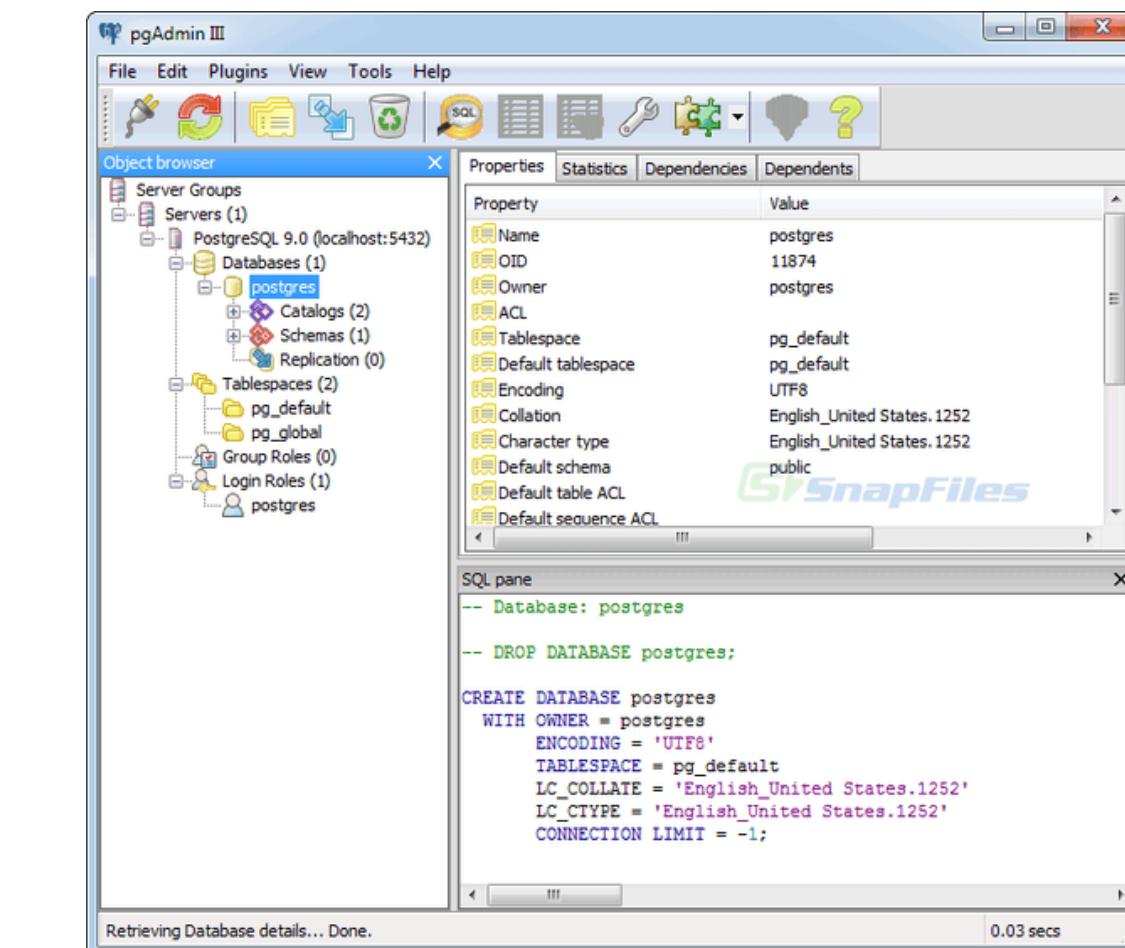
1996

PostgreSQL

- SGDBR de Código Aberto;
- Desenvolvido pela PostGres
- Multiplataforma;
- Sua primeira versão foi lançada em 8 julho de 1996 (26 anos)
- Atualmente se encontra na versão 15.1 com suporte a 28 idiomas



1996 - Atualmente



Interface do pgAdmin do PostgreSQL

1998

Modelo Não Relacional

- Os Bancos de Dados Não Relacional (BnR), existem desde o surgimento do conceito de Banco de Dados, uma vez que os primeiros modelos tinha o conceito de dados desnormalizados.
- Apenas em 1998, por meio Carlo Strozzi, esse conceito oposto ao Relacional, que ele denominou com NoREL (Não Relacional) voltou a ser mencionado na comunidade acadêmica.
- A partir desse novo modelo tinha como caracterísiticas:
 - Escalonamento Horizontal;
 - Esquema Dinâmico (Orientado a Colunas, Chave-Valor, Grafos e/ou Documentos)
 - Flexibilidade no Padrão ACID (Atomicidade, Consistência, Isolamento e Durabilidade)
- Strozzi NoSQL foi o primeiro Sistema Gerencial de Banco de Dados Não Relacional (SGBDnR)

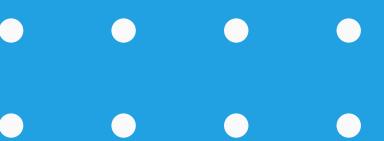


CARLO STROZZI
O CRIADOR DO MODELO NÃO RELACIONAL

90-99

Alguns Pontos Chave sobre a DEC. 90

- Surgimento de novos SGBDs, inclusive com Interface Gráfica ao Usuário
- O nascimento da Internet comercial e as aplicações WEB com a necessidade de armazenar informações - Os primeiros SGBD à Internet
- A Maturidade sobre o conceito Modelo Relacional
- Surgimento de novos conceitos (Data Warehouse, WIIS, GIS, Data Mining)
- A evolução da computação na industria e a necessidade de armazenar informação de maneira centralizada e normalizada (pensando na integridade e a qualidade dos dados)
- O Surgimento dos Bancos de Dados Não Relacional (No-SQL)



2000

SQLite

- É uma Biblioteca que implementa um Banco de dados SQL nativamente (embutido);
- Desenvolvido por Richard Hipp;
- Voltado para aplicações em desktop ou dispositivos de sistema embarcados;
- Sua primeira versão foi lançada em 17 de agosto de 2000 (17 anos)
- Se encontra na versão 3.40



Disponível em:

<https://www.sqlite.org/index.html>



2000

Aplicativos Web com XML

- O grande investimento em empresas de Internet impulsiona as vendas de ferramentas para conexão Web/Internet/BD: *Active Server Pages, Front Page, Java Servlets, JDBC, Enterprise Java Beans.*
- Surgimento da linguagem XML (*Extensible Markup Language*) e o XQUERY para manipulação e exibição de dados em páginas na internet



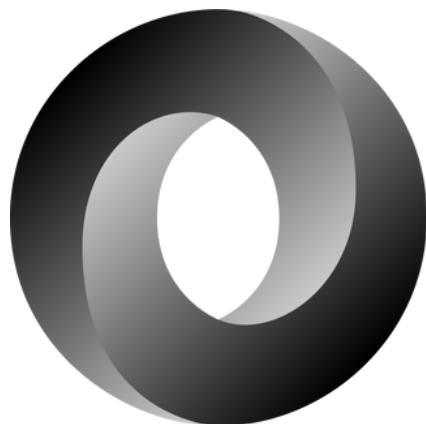
```
<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
- <MUSICAS xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  - <MUSICA>
    <NOME>A Fórmula Do Amor</NOME>
    <CANTOR>Kid Abelha</CANTOR>
    <LETRA>Eu tenho gestos aptos</LETRA>
  </MUSICA>
  - <MUSICA>
    <NOME>A Viagem</NOME>
    <CANTOR>Roupa Nova</CANTOR>
    <LETRA>Há tanto tempo que eu deixei você</LETRA>
  </MUSICA>
  - <MUSICA>
    <NOME>Águas De Março</NOME>
    <CANTOR>Elis Regina</CANTOR>
    <LETRA>É pau é pedra</LETRA>
  </MUSICA>
</MUSICAS>
```

Exemplo XML

2000

Aplicativos Web com JSON

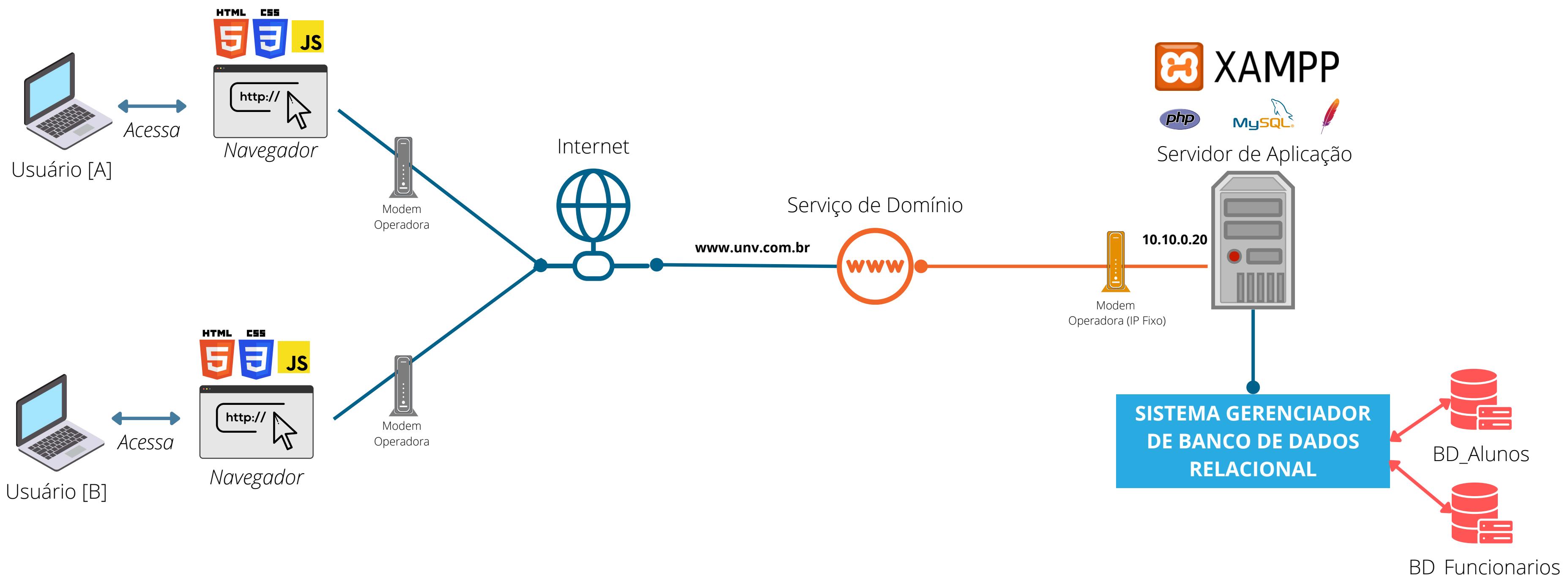
- Em razão da grande dificuldade de manipulação com o XML, **surgiu o formato JSON** (JavaScript Object Notation)
- Tem a principal **função na troca de dados simples e rápida entre os sistemas.**
- Nas aplicações WEB essa ferramento ou tecnologia pode ser aplicado no Front-End (Tecnologias voltadas no Navegador: HTML5, CSS3 e JavaScript) e o Back-End (Tecnologias voltadas no Servidor: PHP, Node.JS, Python, Java e etc.)



```
1 {"Alunos": [
2     { "nome": "Edson Sales Arantes", "notas": [ 8, 9, 5 ] },
3     { "nome": "Luiz Livelli ", "notas": [ 8, 10, 7 ] },
4     { "nome": "Caique Caicedo De Plata", "notas": [ 10, 10, 9 ] }
5 ]}
```

Exemplo JSON

SGBDR depois dos Aplicativos Web



Esse diagrama é baseado sobre os aplicativos web (dinâmicos) que utilizavam o Navegador do Usuário (CSS, JavaScript e HTML) que recebiam informações de um Servidor de Aplicativos (XAMPP com Apache, MySQL e PHP)

2006

ARMAZENAMENTO EM NÚVEM

- O Armazenamento em nuvem é a tecnologia que permite que governos, industrias, Universidade ou usuários comuns em armazenar, manter e acessar dados em servidores de alta disponibilidade via internet.
- Em 2006, a amazon foi primeira empresa a lançar o serviço a núvem para empresas, denominado de Amazon Web Services (AWS)



2006 - 2010



Atualmente

2006

BANCO DE DADOS EM NÚVEM

- Banco de dados em nuvem é um base de dados que normalmente é executado em um computação em nuvem plataforma e o acesso ao banco de dados é fornecido como um serviço, que é denominado Banco de dados como serviço (**DBaaS**).



Existe desde 2006
Oferece **Amazon RDS** como DBaaS



Google Cloud

Existe desde 2008
Oferece **Firebase** como DBaaS



Existe desde 2010
Oferece Azure **SQL DataBase** como DBaaS



IBM Cloud

Existe desde 2011
Oferece **Db2** como DBaaS

ORACLE
Cloud

Existe desde 2016
Oferece **ORACLE** como DBaaS

00-10

Alguns Pontos Chave sobre a DEC. 00

- Surgimento dos primeiros SGBDnR
- O nascimento dos serviços de núvem
- A Maturidade sobre o conceito Modelo Relacional e Não Relacional
- Surgimento do DBaaS
- A evolução da computação na industria e a necessidade de armazenar informação de maneira centralizada e desnormalizada (pensando no empenho)

2009

MariaDB

- SGDBR;
- Desenvolvido pelo fundador do MySQL
- Multiplataforma (Linux, macOS, Windows e etc);
- Sua primeira versão foi lançada em 22 de Janeiro de 2009 (13 anos)
- MariaDB é considerado um *fork* do MySQL, tendo as mesmas funcionalidades e características.
- O MariaDB é considerado um substituto natural (ou sucessor) do MySQL



The screenshot shows the Navicat for MariaDB application window. The left sidebar displays a tree view of database objects, including 'Navicat Cloud', 'Project MH-0015', 'Head Office 305', 'ProjectDT-0052 (marybrown)', 'Production Server', and 'sakila'. The 'Tables' node under 'sakila' is expanded, showing tables like 'actor', 'address', 'category', 'city', 'country', 'customer', 'film', 'film_actor', 'film_category', 'film_text', 'inventory', 'language', 'payment', 'rental', 'staff', and 'store'. The 'customer' table is selected. The main pane displays the structure of the 'customer' table, which has 599 rows and was last modified on 2020-05-22 16:04:49. The table uses InnoDB storage engine and has an auto-increment column. The right pane provides detailed information about the selected table, including its columns, data types, and indexes.

Nom	Lignes	Moteur	Date créée	Date modifiée
actor	200	InnoDB	2017-05-22 16:04:25	2020-05-22 16:04:48
address	603	InnoDB	2017-05-22 16:04:25	2020-05-22 16:04:49
category	16	InnoDB	2017-05-22 16:04:25	2020-05-22 16:04:49
city	600	InnoDB	2017-05-22 16:04:26	2020-05-22 16:04:49
country	109	InnoDB	2017-05-22 16:04:26	2020-05-22 16:04:49
customer	599	InnoDB	2017-05-22 16:04:27	2020-05-22 16:04:49
film	1000	InnoDB	2017-05-22 16:04:27	2020-05-22 16:04:51
film_actor	5462	InnoDB	2017-05-22 16:04:27	2020-05-22 16:04:52
film_category	1000	InnoDB	2017-05-22 16:04:28	2020-05-22 16:04:53
film_text	1000	InnoDB	2017-05-22 16:04:28	2020-05-22 16:04:51
inventory	4581	InnoDB	2017-05-22 16:04:32	2020-05-22 16:04:53
language	6	InnoDB	2017-05-22 16:04:32	2020-05-22 16:04:53
payment	16086	InnoDB	2017-05-22 16:04:33	2020-05-22 16:05:04
rental	16086	InnoDB	2017-05-22 16:04:33	2020-05-22 16:05:08
staff	2	InnoDB	2017-05-22 16:04:33	2020-05-22 16:05:08
store	2	InnoDB	2017-05-22 16:04:34	2020-05-22 16:05:08

Interface

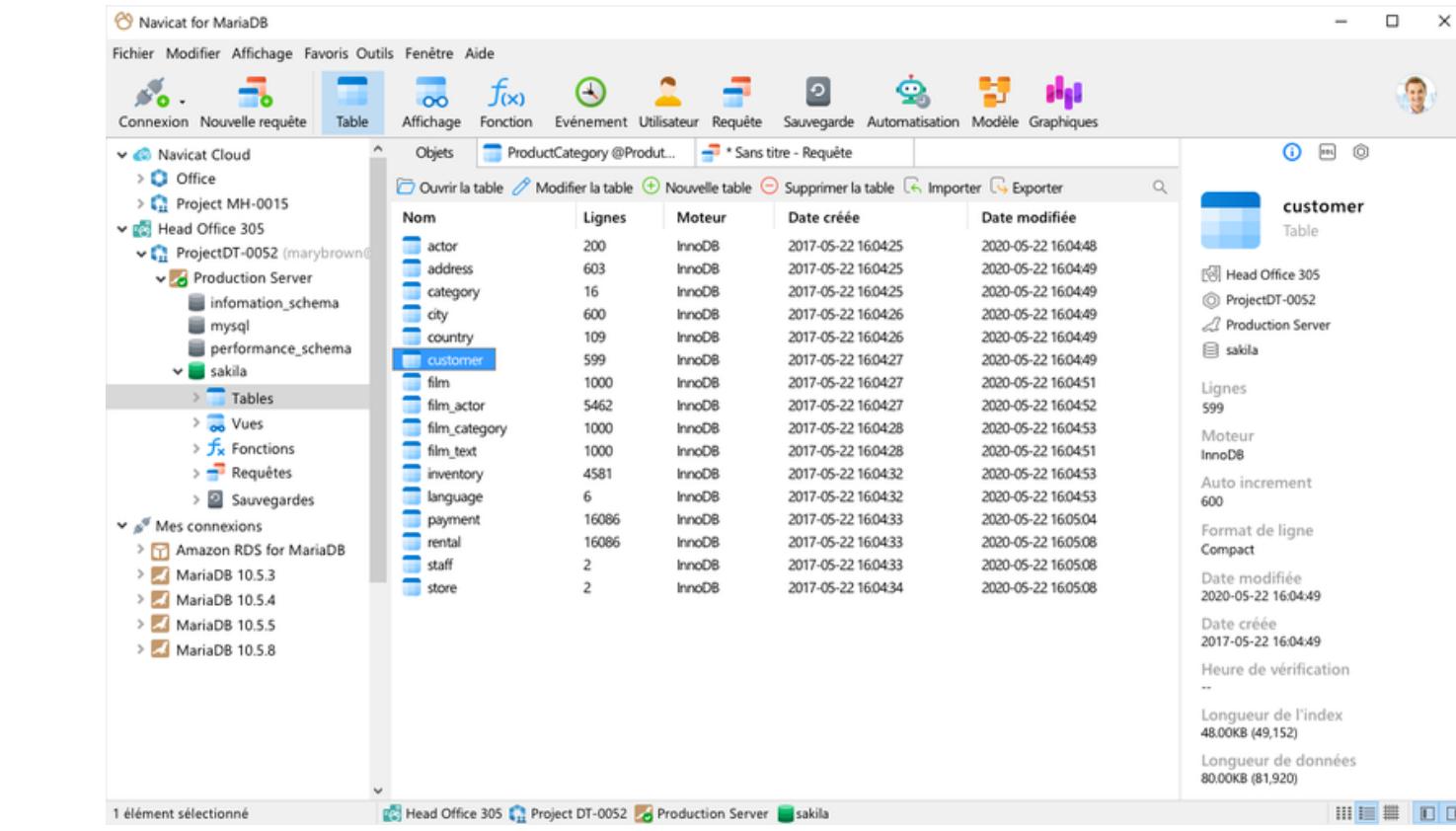
2009

MariaDB

- SGDBR;
- Desenvolvido pelo fundador do MySQL
- Multiplataforma (Linux, macOS, Windows e etc);
- Sua primeira versão foi lançada em 22 de Janeiro de 2009 (13 anos)
- MariaDB é considerado um *fork* do MySQL, tendo as mesmas funcionalidades e características.
- O MariaDB é considerado um substituto natural (ou sucessor) do MySQL



2009- Atualmente

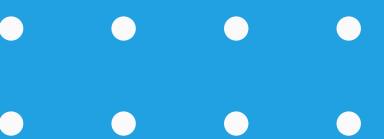


The screenshot shows the Navicat for MariaDB software interface. The main window displays the 'customer' table from the 'sakila' database. The table has the following structure:

Nom	Lignes	Moteur	Date créée	Date modifiée
actor	200	InnoDB	2017-05-22 16:04:25	2020-05-22 16:04:48
address	603	InnoDB	2017-05-22 16:04:25	2020-05-22 16:04:49
category	16	InnoDB	2017-05-22 16:04:25	2020-05-22 16:04:49
city	600	InnoDB	2017-05-22 16:04:26	2020-05-22 16:04:49
country	109	InnoDB	2017-05-22 16:04:26	2020-05-22 16:04:49
customer	599	InnoDB	2017-05-22 16:04:27	2020-05-22 16:04:49
film	1000	InnoDB	2017-05-22 16:04:27	2020-05-22 16:04:51
film_actor	5462	InnoDB	2017-05-22 16:04:27	2020-05-22 16:04:52
film_category	1000	InnoDB	2017-05-22 16:04:28	2020-05-22 16:04:53
film_text	1000	InnoDB	2017-05-22 16:04:28	2020-05-22 16:04:51
inventory	4581	InnoDB	2017-05-22 16:04:32	2020-05-22 16:04:53
language	6	InnoDB	2017-05-22 16:04:32	2020-05-22 16:04:53
payment	16086	InnoDB	2017-05-22 16:04:33	2020-05-22 16:05:04
rental	16086	InnoDB	2017-05-22 16:04:33	2020-05-22 16:05:08
staff	2	InnoDB	2017-05-22 16:04:33	2020-05-22 16:05:08
store	2	InnoDB	2017-05-22 16:04:34	2020-05-22 16:05:08

The interface also shows the 'customer' table selected in the right panel, with details such as 599 rows, InnoDB engine, and auto-increment for the 'customer_id' column.

Interface





2019 ORACLE 19c

19c ORACLE®
Database

Disponível em:
<https://www.oracle.com/>



The screenshot shows the Oracle Live SQL interface. At the top, there is a navigation bar with icons for Home, Recent, Database, Schema, and Utilities. The main area is titled "SQL Worksheet". A SQL statement is entered in the worksheet:

```
1 select * from v$version;
```

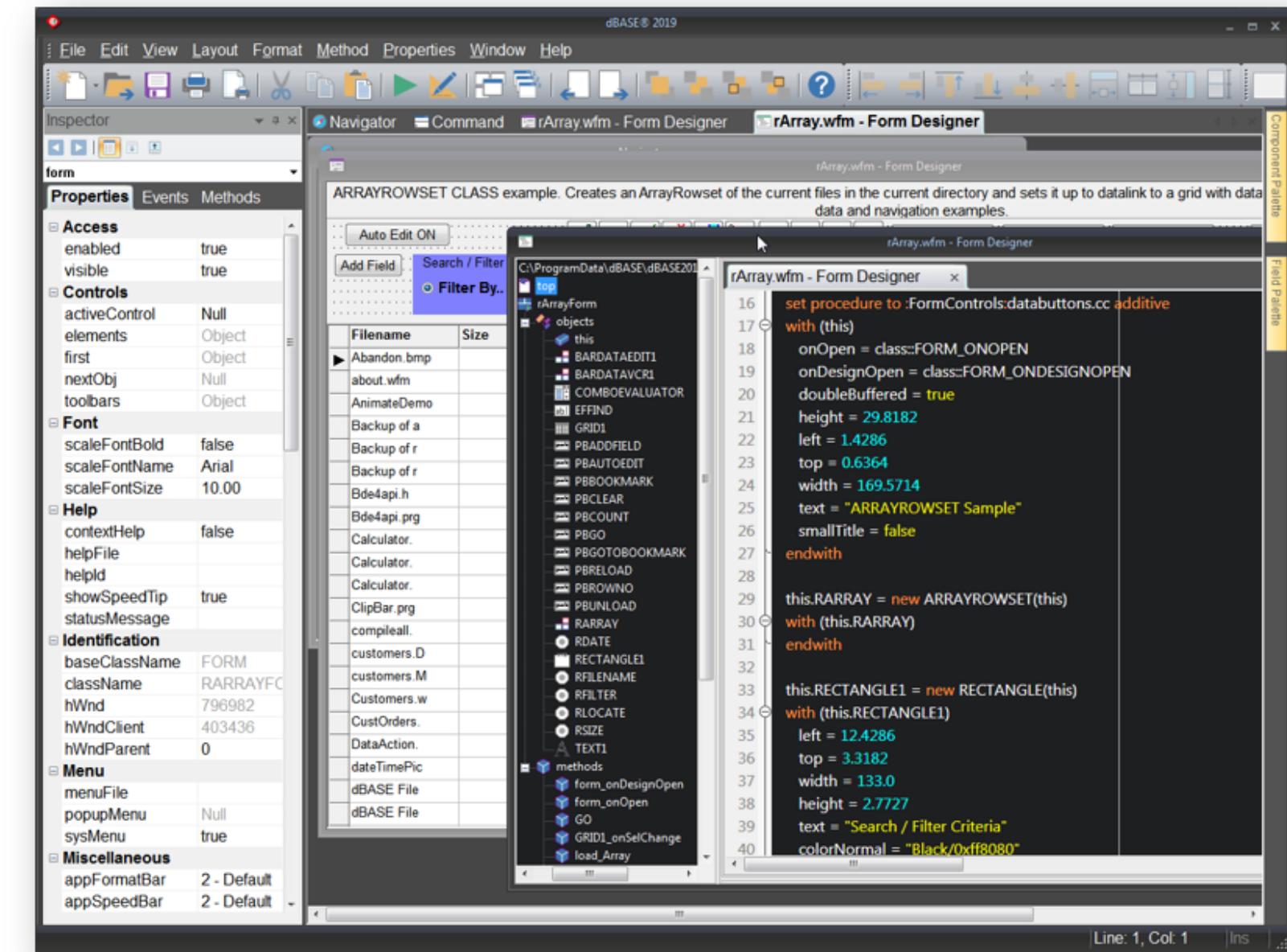
The results are displayed in a table:

BANNER	BANNER_FULL	BANNER_LEGACY	CON_ID
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production	Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production Version 19.2.0.0.0	Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production	0

At the bottom of the interface, there are footer links: © 2019 Oracle Corporation · Privacy · Terms of Use, Oracle Learning Library · Oracle Database Documentation 19c , 18c , 12c · Follow on Twitter, Live SQL 19.1.2, running Oracle Database 19c Enterprise Edition - 19.2.0.0.0, and Built with ❤ using Oracle APEX.

2019

dBASE 2019



Disponível em:

<https://www.dbase.com/>

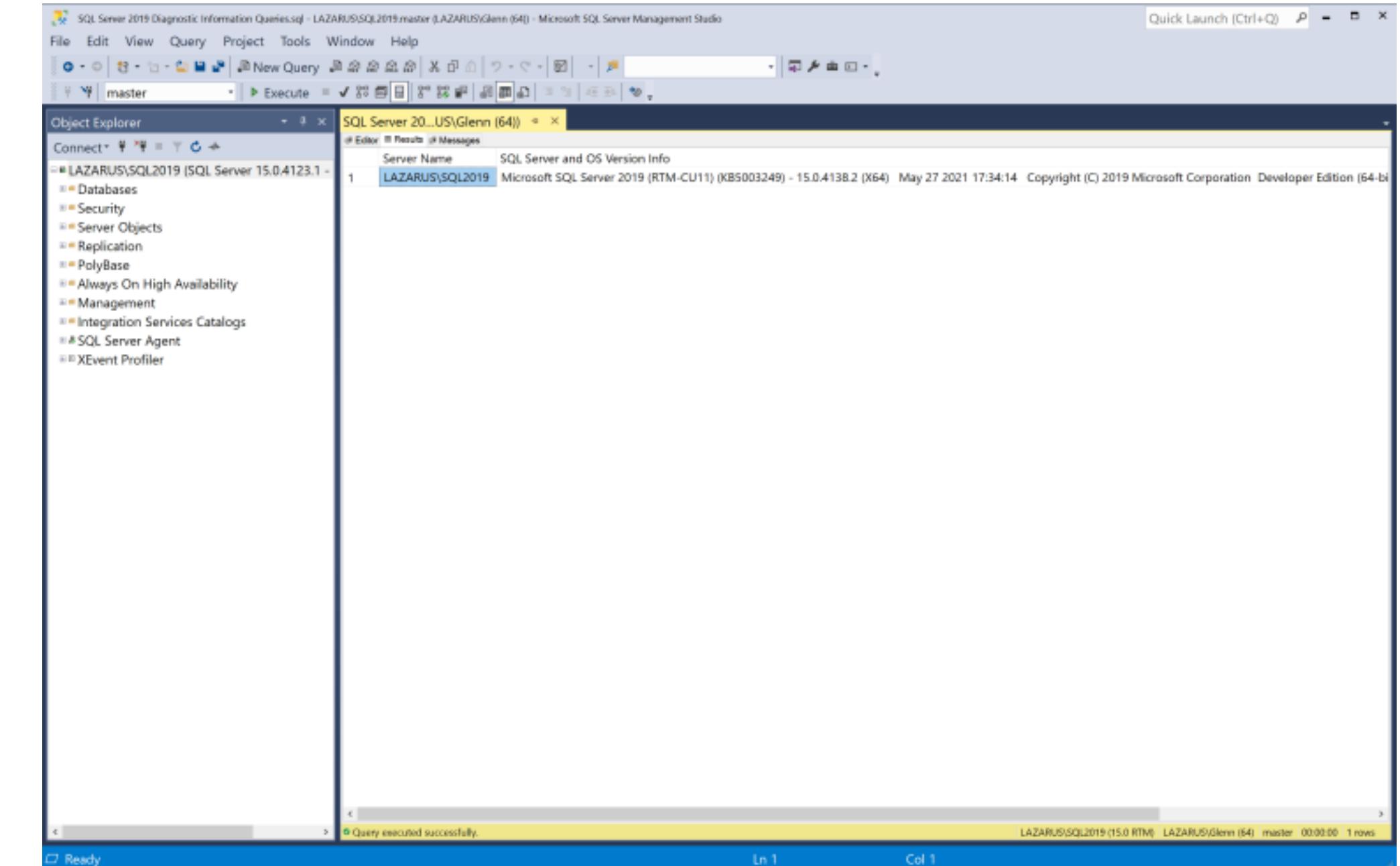


2019

Microsoft SQL Server 2019

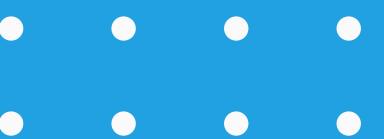


Microsoft®
SQL Server®



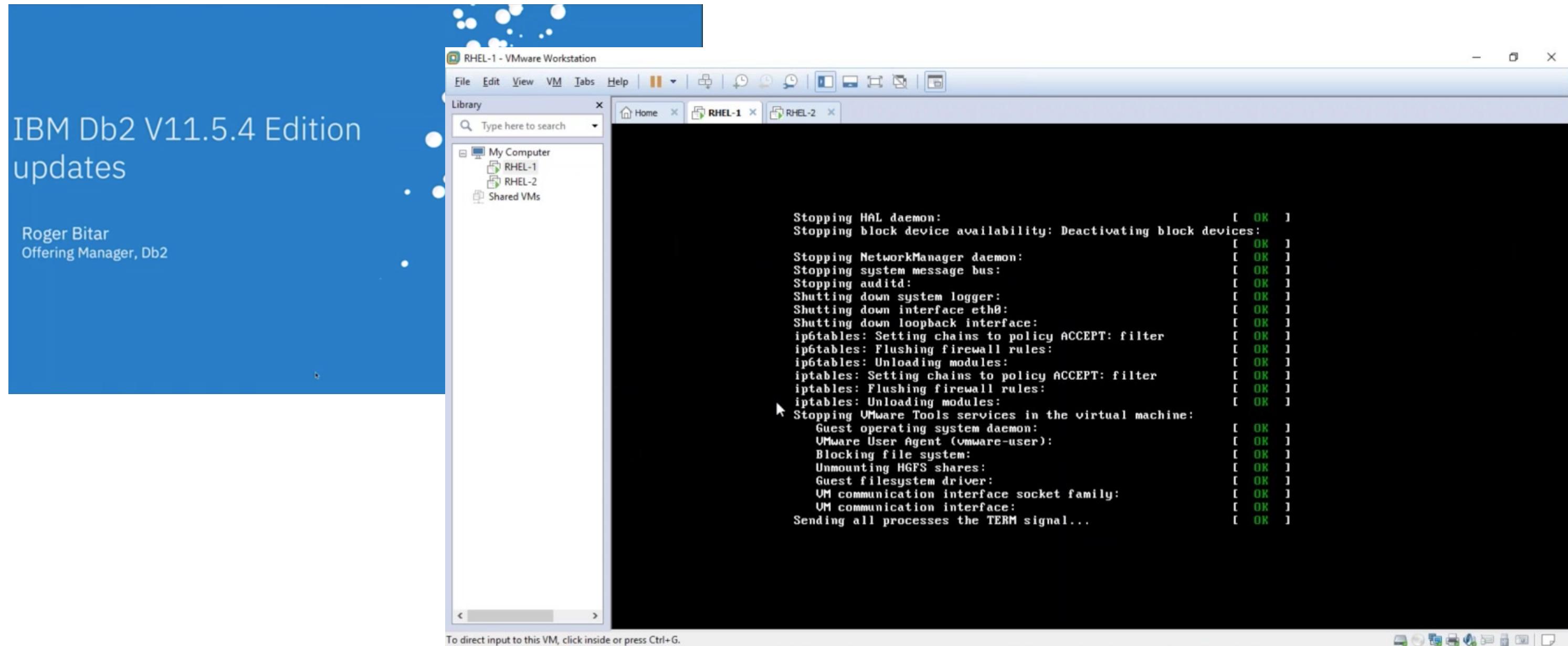
Disponível em:

<https://www.microsoft.com/pt-br/sql-server/sql-server-2019>



2020

IBM Db2 v11.5



Disponível em:

<https://www.ibm.com/products/db2>



2022

PostgreSQL 15.1



PostgreSQL

The screenshot shows the pgAdmin 4 interface. On the left, the 'Browser' pane displays a tree view of database objects under 'PostgreSQL 13' and 'softpedia'. In the center, a 'Create - Database' dialog box is open, showing the SQL code for creating the database:

```
1 CREATE DATABASE "My-database"
2   WITH
3     OWNER = pg_signal_backend
4     TEMPLATE = postgres
5     ENCODING = 'UTF8'
6     LC_COLLATE = 'English_United States.1252'
7     LC_CTYPE = 'POSIX'
8     TABLESPACE = pg_default
9     CONNECTION LIMIT = -1;
10
11 COMMENT ON DATABASE "My-database"
12   IS 'Lorem ipsum';
```

On the right, a status bar shows performance metrics: Rows Fetched: 10,783, Tuples inserted: 791, Tuples updated: 78, and Tuples: 1.

Disponível em:
<https://www.postgresql.org/>





2022

MySQL 8.0



The screenshot shows the MySQL Workbench interface. In the top navigation bar, 'Local' is selected. The left sidebar displays the 'SCHEMAS' tree with 'sakila', 'sys', and 'world' databases. The main area shows the results of the SQL query 'SHOW ENGINES'. The results are presented in a grid table:

Engine	Support	Comment	Transactions	XA	Savepoints
MEMORY	YES	Hash based, stored in memory, useful for temp...	NO	NO	NO
MRG_MYISAM	YES	Collection of identical MyISAM tables	NO	NO	NO
CSV	YES	CSV storage engine	NO	NO	NO
FEDERATED	NO	Federated MySQL storage engine	HULL	HULL	HULL
PERFORMANCE_SCHEMA	YES	Performance Schema	NO	NO	NO
MyISAM	YES	MyISAM storage engine	NO	NO	NO
InnoDB	DEFAULT	Supports transactions, row-level locking, and fo...	YES	YES	YES
BLACKHOLE	YES	/dev/null storage engine (anything you write to ...	NO	NO	NO
ARCHIVE	YES	Archive storage engine	NO	NO	NO

Disponível em:
<https://www.mysql.com/>

2023

Atualmente [...]

- Atualmente existem vários modelos de bancos de dados tais como, orientado a objetos, orientado a documentos, orientado a grafos, ETC . Porém, **o mais comum ainda é o banco de dados relacional.**
- **A decisão entre qual modelo de banco de dados utilizar baseia-se no tipo de dados que você pretende armazenar.**
- Data mining (Mineração de dados), data warehouse (Armazém de dados), data analysis (Análise de dados) são técnicas utilizadas atualmente e no futuro serão utilizados cada vez mais.

Alguma Dúvida?



MERCADO
DE TI



PROFISSÕES



Administrador de Banco de Dados (DBA)

De modo geral sua função é realizar a implementação, manutenção e o gerenciamento de bancos de dados. Além disso, é responsável em garantir que os sistemas de banco de dados estejam funcionando corretamente e estejam atualizados com as últimas tecnologias.



Analista/Cientista de Dados

É responsável em coletar, compilar, analisar e interpretar os dados, que poderão ser usados para promover um negócio, projeto ou pesquisa e, criar novos produtos ou estratégias.

Vagas

92

na cidade de São Paulo
Jan/2023

73

na cidade de São Paulo
Jan/2023

18

na cidade de São Paulo
Jan/2023



Salários de DBA em São Paulo

Atualizado em 30 de jan. de 2023

Confiança muito alta

R\$ 8.640 /mês

Média salarial

433 salários



Outras remunerações ①

Média: R\$ 13.965 Intervalo: R\$ 425 - R\$ 42.249

DBA

✓ Média Salarial

R\$ 7.987,26

✉ Vagas na Catho

Buscar

Salário-base médio ?

R\$ 8.269

Por mês

Igual

dentro da média nacional

A faixa salarial para o cargo de analista de dba é de R\$ 8.269 por mês para São Paulo, SP. 4 salários informados. Atualização em 17 de janeiro de 2023

Vagas

208

na cidade de São Paulo
Jan/2023

48

na cidade de São Paulo
Jan/2023

29

na cidade de São Paulo
Jan/2023



Salários de Cientista De Dados em São Paulo

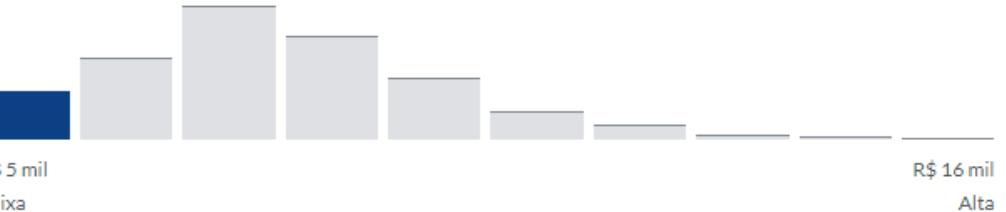
Atualizado em 29 de jan. de 2023

Confiança muito alta

R\$ 9.148 /mês

Média salarial

988 salários



Outras remunerações ^①

Média: R\$ 20.000

Intervalo: R\$ 5.677 - R\$ 63.840



Cientista de Dados

Média Salarial

R\$ 5.280,45

Vagas na Catho

Buscar

Salário-base médio ?

R\$ 4.610

Por mês ▼

↓ 10%

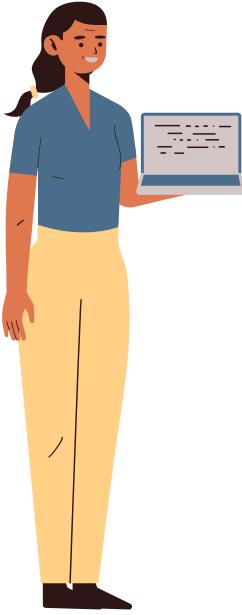
abaixo da média nacional

A faixa salarial para o cargo de cientista de dados é de R\$ 4.610 por mês para São Paulo, SP. 33 salários informados. Atualização em 20 de janeiro de 2023





PROFISSÕES



Analista de Business Intelligence (BI)

Montar os modelos de negócio, levantamento de requisitos, criação de cubos de dados e templates de relatórios que precisarão abordar todos os cenários previamente definidos, após coleta, organização e análise das informações de mercado que dão suporte à gestão do negócio.

Vagas

1015

na cidade de São Paulo
Jan/2023

131

na cidade de São Paulo
Jan/2023

989

na cidade de São Paulo
Jan/2023

Salários de Analista De BI em São Paulo

Atualizado em 30 de jan. de 2023

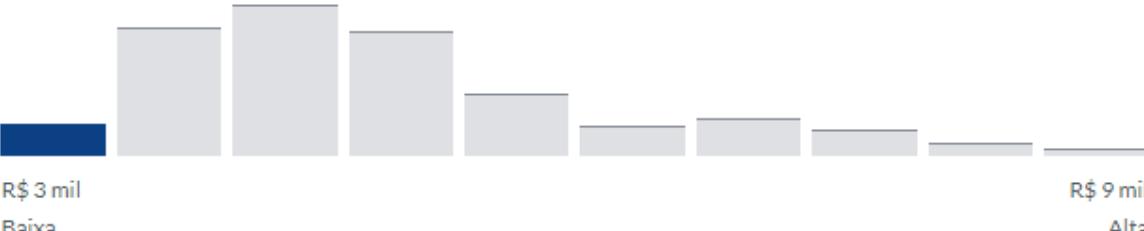


Confiança muito alta

R\$ 4.920/mês

Média salarial

358 salários



Analista de Business Intelligence

✓ Média Salarial

R\$ 4.102,01

■ Vagas na Catho

Buscar



Salário-base médio ?

R\$ 4.819

Por mês ▾

↑ 11%

acima da média nacional

A faixa salarial para o cargo de analista de bi é de R\$ 4.819 por mês para São Paulo, SP. 118 salários informados. Atualização em 24 de janeiro de 2023



PROFISSÕES



Analista de Sistemas/Suporte

Realiza seleção, implantação e manutenção de hardwares e softwares básico e de apoio, define controle de acesso aos recursos, participa na documentação de rotina e acompanha o desempenho dos recursos técnicos instalados.



Desenvolvedor PL/SQL

Responsável por criar e melhorar os programas de banco de dados utilizando a linguagem de programação PL/SQL. Normalmente o mesmo é capaz em criar e manter tabelas, procedimentos armazenados, índices, gatilhos; solucionar problemas de desempenho; e otimizar consultas.

Vagas

108

na cidade de São Paulo
Jan/2023

94

na cidade de São Paulo
Jan/2023

22

na cidade de São Paulo
Jan/2023

Salários de Desenvolvedor PL/SQL em São Paulo

Atualizado em 4 de jan. de 2023



Confiança muito alta

R\$ 5.800/mês

Média salarial

17 salários



Outras remunerações ①



Analista Programador SQL

↑ Média Salarial

R\$ 2.734,63

Vagas na Catho

Buscar



Salário-base médio ?

R\$ 5.907

Por mês ▼

↑ 40%

acima da média nacional

A faixa salarial para o cargo de dba sql server é de R\$ 5.907 por mês para São Paulo, SP. 17 salários informados. Atualização em 12 de janeiro de 2023

Vagas

108

na cidade de São Paulo
Jan/2023

223

na cidade de São Paulo
Jan/2023

669

na cidade de São Paulo
Jan/2023

Salários de Analista De Suporte em São Paulo

Atualizado em 30 de jan. de 2023



Confiança muito alta

R\$ 2.457 /mês

Média salarial

3.151 salários



Analista de Suporte Técnico

↗ Média Salarial

R\$ 2.317,37

Vagas na Catho

Buscar



Salário-base médio ?

R\$ 2.630

Por mês ▼

↑ 19%

acima da média nacional

A faixa salarial para o cargo de analista de suporte técnico é de R\$ 2.630 por mês para São Paulo, SP. 95 salários informados. Atualização em 25 de janeiro de 2023

Atividade

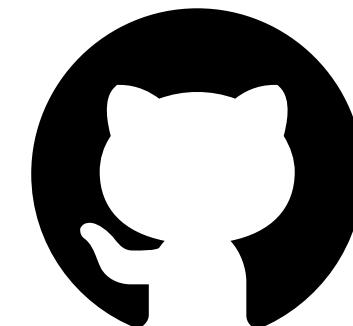
A necessidade do mercado na busca de profissionais de T.I, leva muitas empresas em engajar as suas vagas em diferentes sites ou redes sociais específicos de trabalho.

Porém muitos profissionais deixam essas oportunidades passarem por não ter um bom *networking* e um portfólio digital (galeria de produtos e serviços executados por ele).

Portanto, essa atividade tem como objetivo em você (estudante) em criar e preencher o seu perfil no:



LinkedIn



GitHub

Atividade

Prazo: 11/03/2023

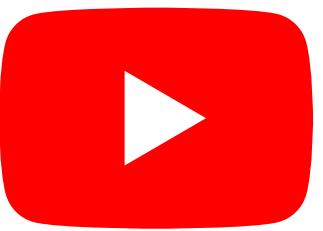
Nota: 0 a 10 (compondo a nota de atividades da N1 (40%))

A Não entrega da atividade pelo o Aluno no prazo estipulado será pontuado 0 pts

Atividade Liberada



Conteúdo Extra



Episódio 025: Banco de Dados

Hoje vamos falar daquele lugar mágico onde o nosso trabalho mora: o banco de dados! A rainha dos bancos de dados Dani Monteiro veio trocar uma ideia com os...

Google Podcasts /

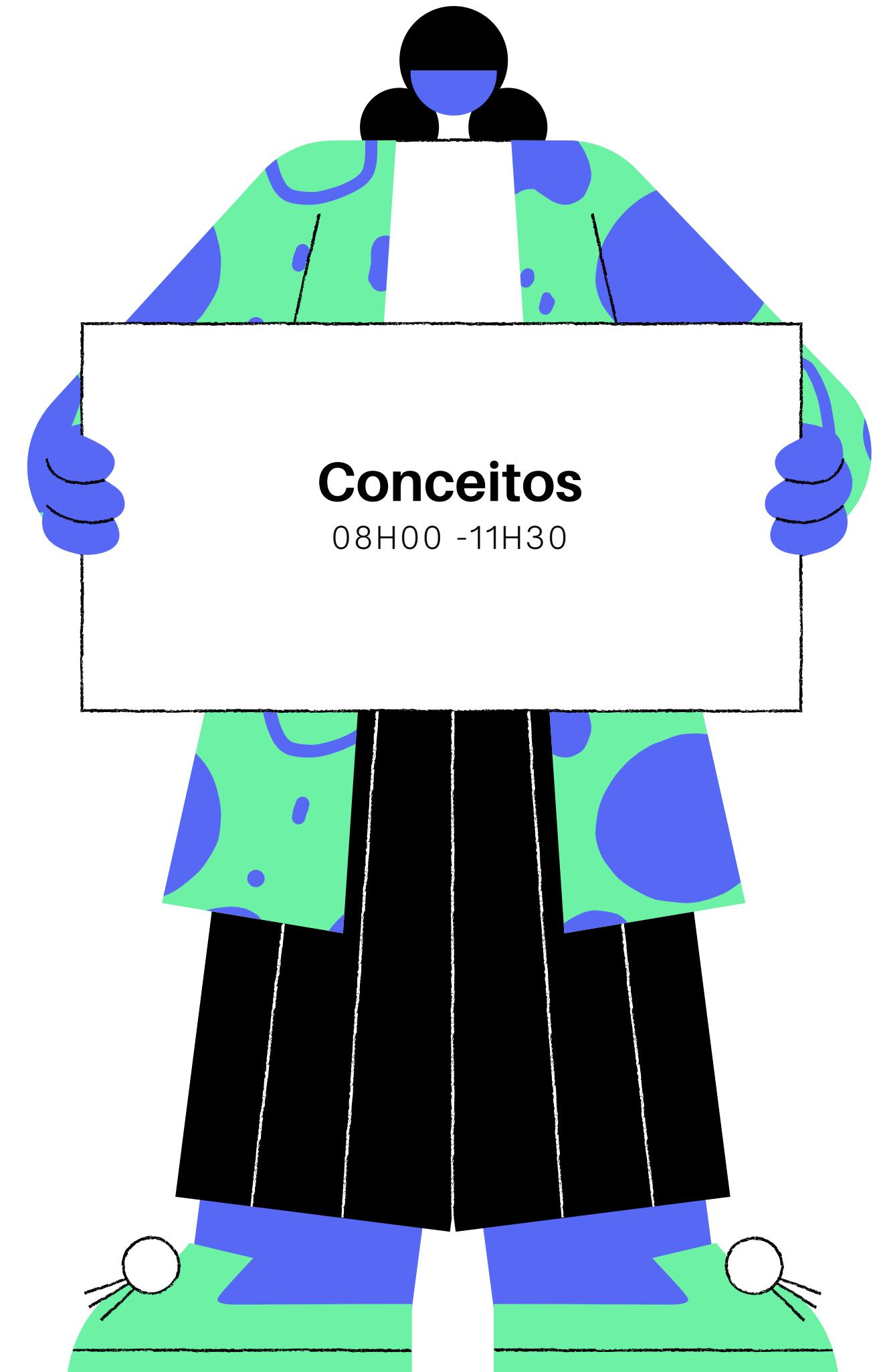


Banco de Dados Relacional

Banco de Dados Relacional, chave primária, estrangeira e SQL

Google Podcasts /

Próxima Aula



REFERÊNCIAS

- [HTTPS://WWW.ORACLE.COM/BR/DATABASE/WHAT-IS-DATABASE/](https://www.oracle.com/br/database/what-is-database/)
- [HTTPS://MUNDODELIVROS.COM/EPOPEIA-DE-GILGAMESH/](https://mundodelivros.com/eopeia-de-gilgamesh/)
- [HTTPS://CHC.ORG.BR/ACERVO/QUAL-FOI-O-PRIMEIRO-LIVRO/](https://chc.org.br/acervo/qual-foi-o-primeiro-livro/)
- [HTTPS://WWW.SEAS.UPENN.EDU/~ZIVES/03F/CIS550/CODD.PDF](https://www.seas.upenn.edu/~zives/03f/cis550/codd.pdf)
- [HTTPS://DL.ACM.ORG/DOI/10.1145/320434.320440](https://dl.acm.org/doi/10.1145/320434.320440)
- [HTTPS://TECNOBLOG.NET/RESPONDE/O-QUE-E-TCP-IP/](https://tecnoblog.net/responde/o-que-e-tcp-ip/)
- [HTTPS://DSF.BERKELEY.EDU/CS262/SYSTEMR-ANNOTATED.PDF](https://dsf.berkeley.edu/cs262/systemr-annotated.pdf)
- [HTTPS://HISTORIADOCOMPUTADOR.COM/UNIVAC-O-PRIMEIRO-COMPUTADOR-COMERCIAL/](https://historiadocomputador.com/univac-o-primeiro-computador-comercial/)
- [HTTPS://PT.WIKIPEDIA.ORG/WIKI/ST-506](https://pt.wikipedia.org/wiki/ST-506)
- [HTTPS://WWW.MYSQL.COM/](https://www.mysql.com/)
- [HTTPS://WWW.IBM.COM/PRODUCTS/DB2](https://www.ibm.com/products/db2)
- [HTTPS://WWW.POSTGRESQL.ORG/](https://www.postgresql.org/)
- [HTTPS://WWW.ORACLE.COM/](https://www.oracle.com/)
- [HTTPS://WWW.DEVMEDIA.COM.BR/A-HISTORIA-DOS-BANCO-DE-DADOS/1678](https://www.devmedia.com.br/a-historia-dos-banco-de-dados/1678)
- ELMASRI, R.; NAVATHE, S. B. SISTEMAS DE BANCO DE DADOS. 7 ED. SÃO PAULO: PEARSON, 2018.
- HEUSER, CARLOS ALBERTO. PROJETO DE BANCO DE DADOS 6^a ED. PORTO ALEGRE: BOOKMAN. 2009



Copyright © 2023 Profº Drº Francisco Douglas Lima Abreu

Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito ao autor