



Transformando o futuro das pessoas
e as pessoas para o futuro.

#Senacfaz75



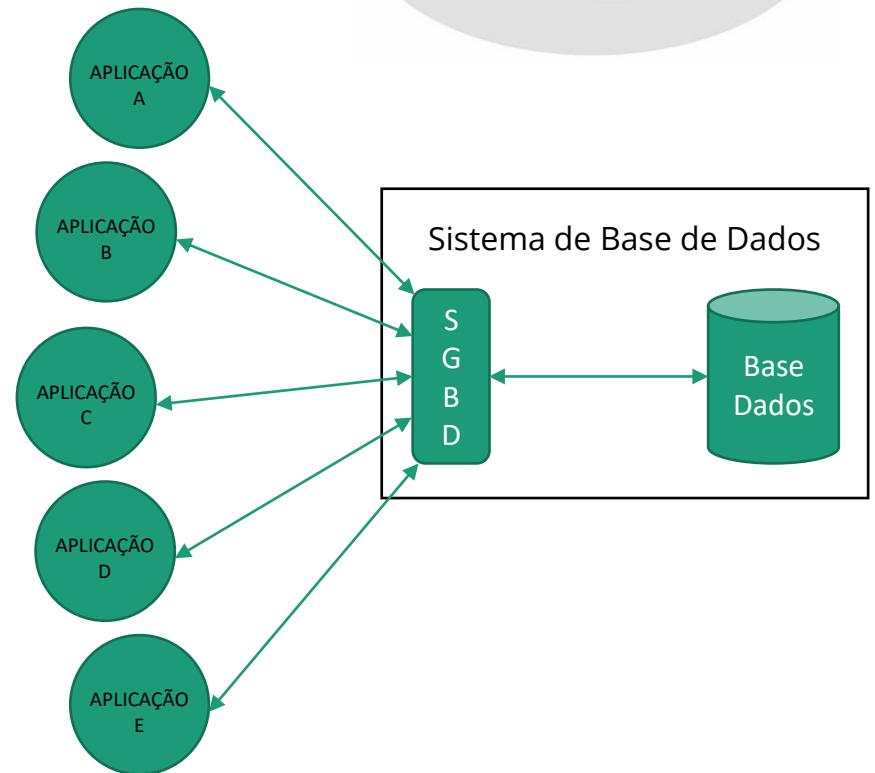
Fundamentos de Banco de Dados e linguagem SQL



O que é um banco de dados?

Um banco de dados ou base de dados nada mais é do que uma coleção de dados inter-relacionados, com algum significado inerente, isto é, informações de interesse de uma ou mais organizações. Ele é projetado, construído e preenchido com dados para um propósito específico de uma ou mais organizações e destinado à utilização por grupo de usuários, diretamente ou por meio de aplicações pré-concebidas.

Podemos também ver um banco de dados como um conjunto de arquivos estruturados de forma a facilitar o acesso aos conjuntos de dados armazenados. Esses arquivos encontram-se, de alguma forma, relacionados. Por exemplo, em um banco de dados escolar podemos encontrar alguns arquivos, tais como: dados dos alunos (nome, endereço, CPF, sexo), dos cursos propriamente ditos (título, ano de lançamento, formato, etc.) e dados sobre as disciplinas (nome, ementa, código, etc.). Para obter informações sobre uma dada disciplina e os alunos que participam dela, como nome da disciplina e nome dos alunos, será necessário consultar os dois primeiros arquivos, que devem estar relacionados. Desta forma, o relacionamento entre os arquivos é uma das condições para que tenhamos um banco de dados, pois somente através destes relacionamentos, teremos a informação propriamente dita.



Entretanto, apenas com a utilização de arquivos relacionados ainda é muito difícil de se acessar estas informações. Dado o acréscimo do volume e dos tipos de dados nas empresas, há uma necessidade de se utilizar softwares especiais para gerenciar as informações, e esses softwares são denominados SGBDs (Sistemas Gerenciadores de Banco de Dados).

Os sistemas gerenciadores de bancos de dados são softwares que manipulam, com eficiência grandes coleções de informações estruturadas e armazenadas de uma forma consistente e integrada. Estes sistemas possuem módulos para consulta, atualização e as interfaces entre o sistema e o usuário. Podemos afirmar, então, que um SGBD é constituído por um conjunto de dados associados a um conjunto de programas para acesso a eles [SILBERSCHATZ, 2006].

Utilidade:

- Sistema acadêmico
- Agências Bancárias
- Rede Hoteleira
- Bibliotecas digitais
- Redes de supermercado etc

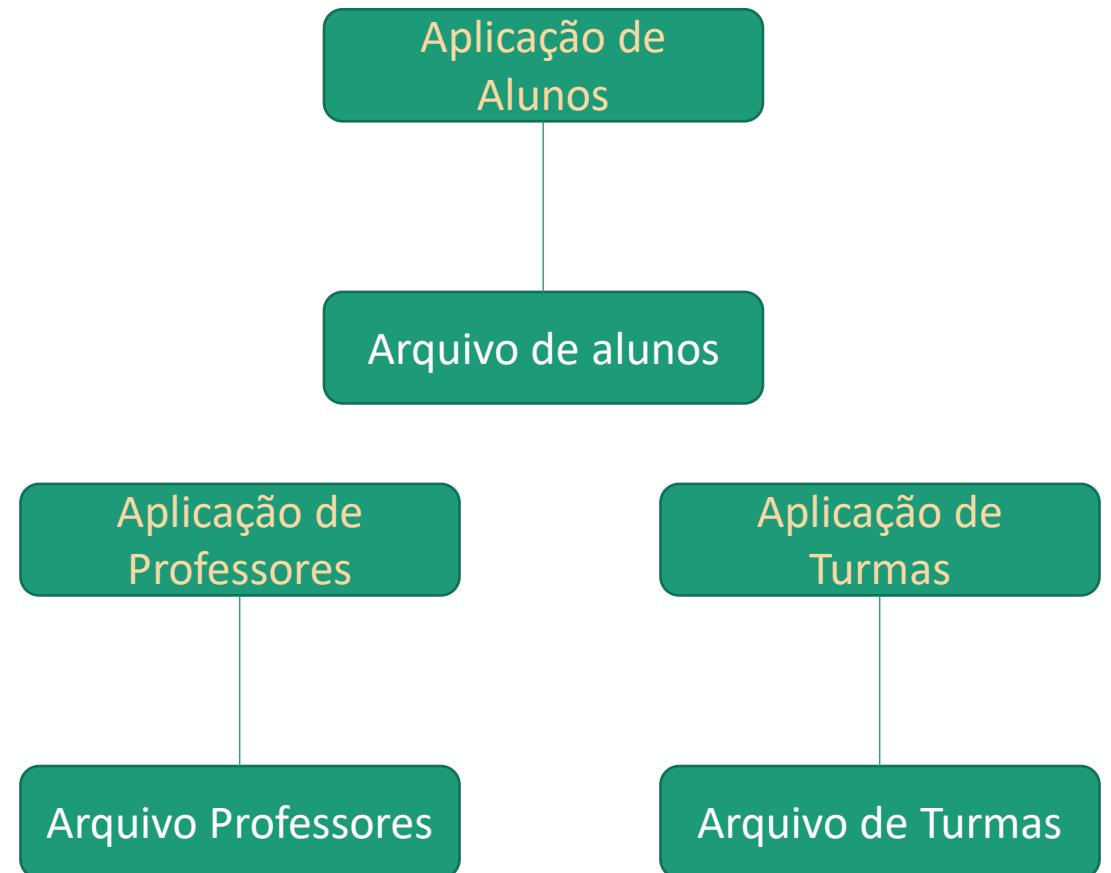
Exemplos de Sistemas Gerenciadores de Banco de Dados



Para saber mais sobre cada um deles, faça uma pesquisa para conhecê-los mais a fundo.

Problemas com a não utilização de um Sistema de Gerenciamento de Banco de Dados

Se não houvesse tais sistemas, nossas informações poderiam ficar armazenadas em um conjunto muito grande de arquivos e sem ligação entre si. Assim os dados de uma aplicação não estariam integrados com os dados de outra aplicação, poderia haver redundância, dados inconsistentes, ou seja, tem um valor em um arquivo e pode ter valor diferente em outro. Não teríamos uma integralidade de dados, veja a figura abaixo:



Problema 1: Redundância

Informações sendo repetidas em vários arquivos. Sendo assim poderíamos ter uma informação dentro de aluno com um valor e a mesma informação dentro de turma com um valor diferente, ou seja, gerando inconsistência em toda a estrutura.

Problema 2: Dificuldade de Acesso

Sistemas de arquivos não possuem ambiente de recuperação de dados armazenados.

Problema 3: Isolamento de dados

Como em cada um dos arquivos que armazenam dados de formas diferentes, fica difícil a construção de uma aplicação que converse com tantas informações guardadas sem uma regra de armazenamento.

Problema 4: Atomicidade

Em SGBDs essa é uma regra de ouro, uma coisa é ou não é! Não há espaço para o meio termo. Eu faço tudo quando estiver tudo certo ou não faço nada. Isso não existe em sistemas de arquivos, pois em seu armazenamento não há algum tipo de controle. Exemplo: transferência bancária.

Aplicação de Alunos

Arquivo de alunos

Aplicação de Professores

Arquivo Professores

Aplicação de Turmas

Arquivo de Turmas

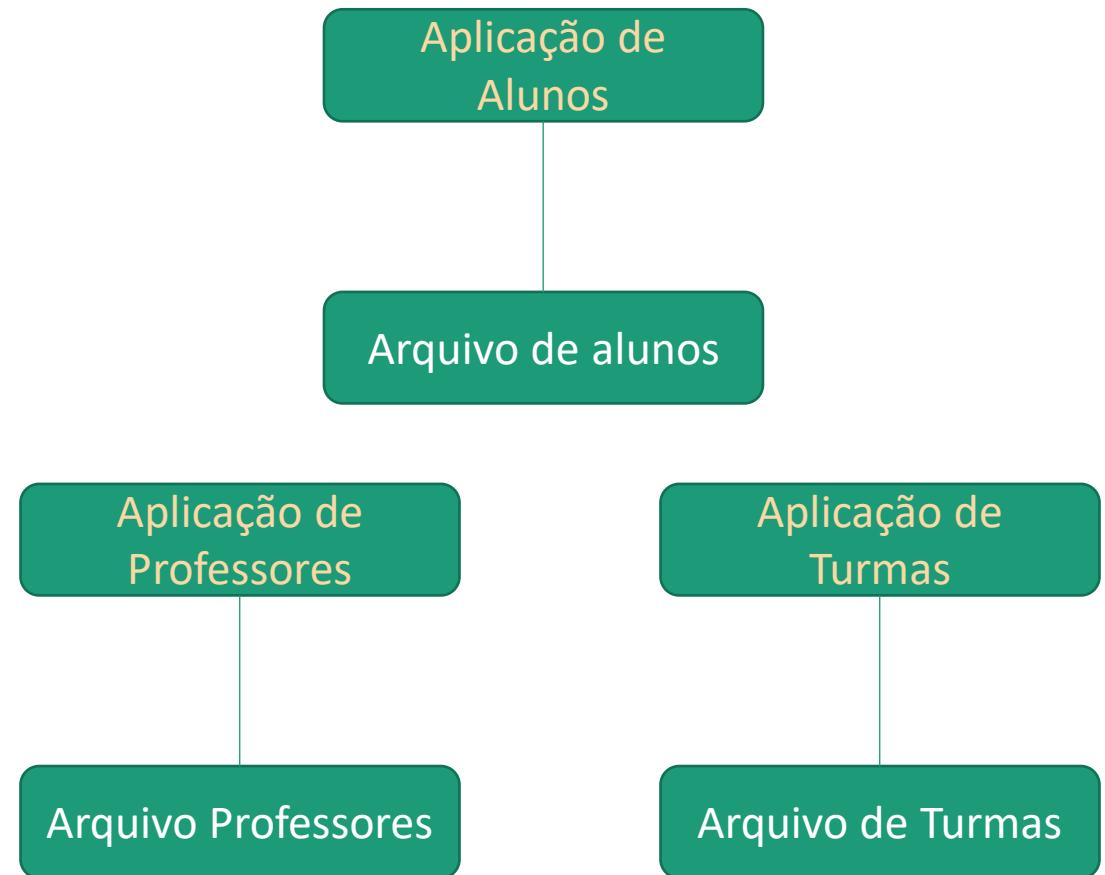
Problema 5: Concorrência de acesso

Em sistemas de arquivos não temos controle de quem está acessando e o quê simultaneamente. veja exemplo:

Tomemos como exemplo que uma conta conjunta A de fundo de formatura- com saldo igual a R\$ 1000,00 - foi acessada de forma simultânea pelos alunos Carolina e Pedro. Carolina sacou R\$100,00 para pagar os convites e Pedro, R\$200,00 para comprar refrigerantes. Pergunta-se: qual o saldo da conta após os saques? Se ambos leram o valor do saldo igual a R\$1000,00, podemos ter como possíveis valores : R\$900,00, R\$800,00, levando-se em conta qual valor foi escrito por ultimo. Nesse caso, nenhum dos dois valores seria valido. O correto seria ter um saldo igual a R\$700,00 e o banco de dados garante isso, mas arquivos isolados podem não garantir. (BRAGA, 2013. P.11.

Problema 6: Segurança

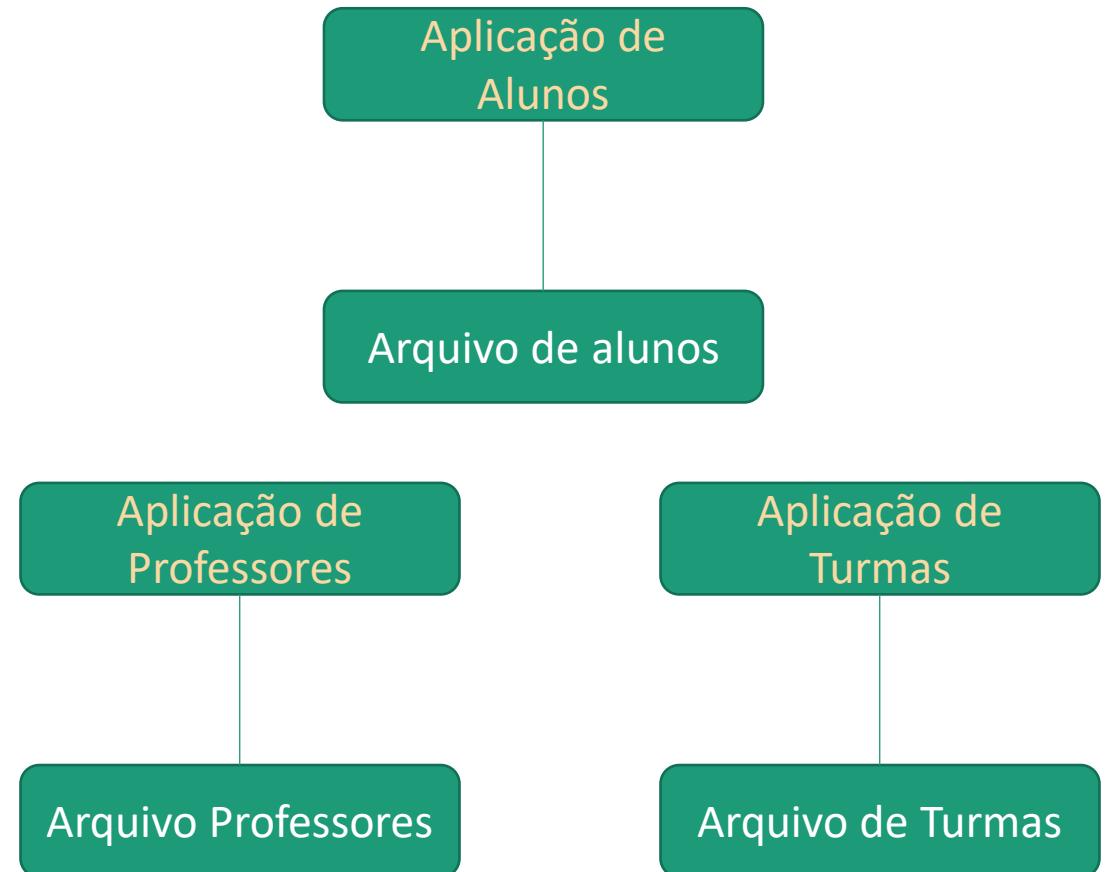
Será que todos os usuários tem perfil para acessar a base de dados em sistemas de arquivos? Imaginemos um usuário tendo total acesso aos dados de um professor em um sistema acadêmico, com acesso a cpf, salários etc. Não seria no mínimo politicamente correto.



Problema 7: Integridade

Vejamos um caso peculiar:

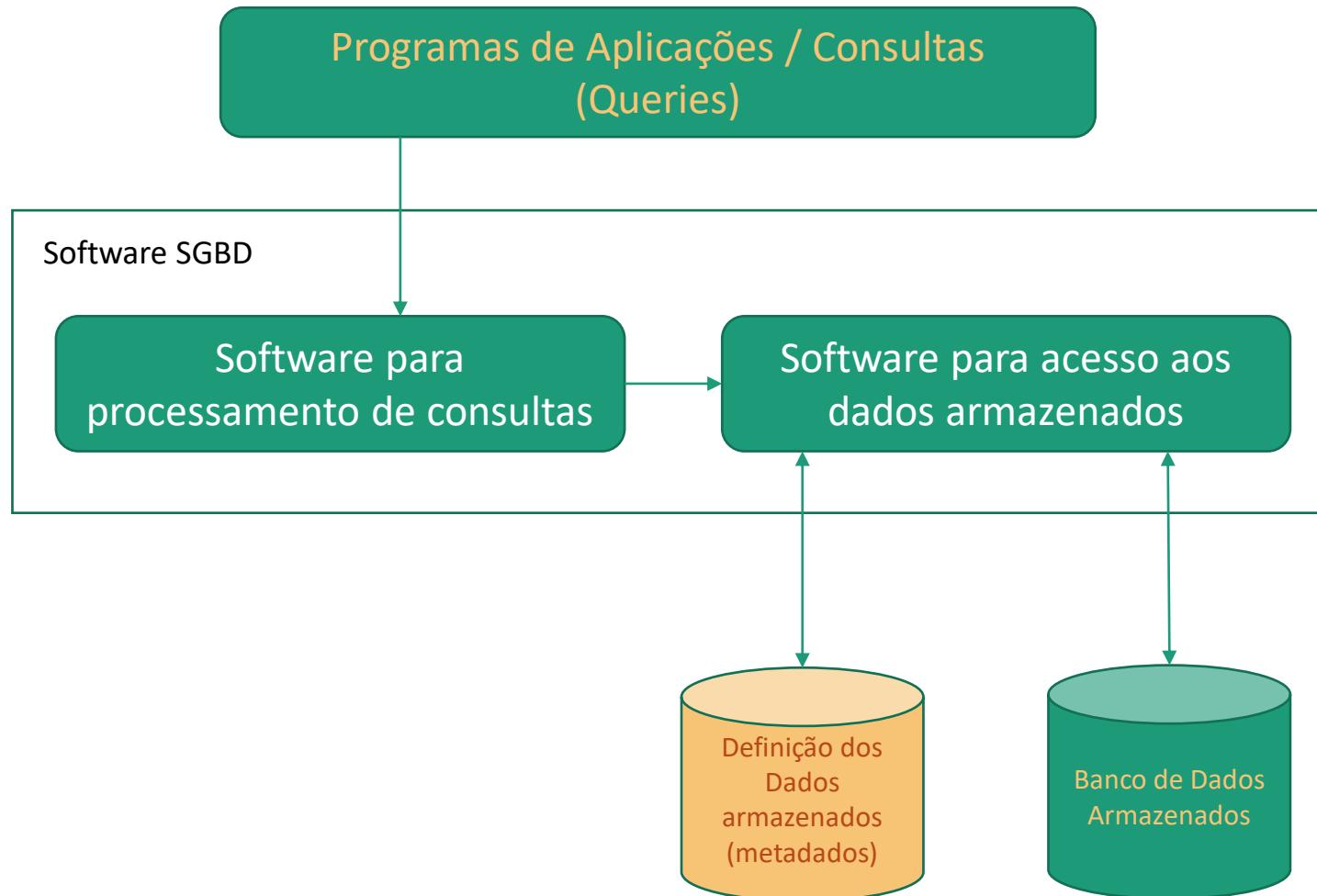
Imaginem o seguinte problema, estamos cadastrando todos os alunos de uma dada turma e, alguns alunos não tem CPF próprio e usam o CPF da mãe. Imaginem agora que temos dois alunos gêmeos e eles vão ser cadastrados usando o mesmo CPF e estamos utilizando o CPF para identificação dos alunos. O que teríamos neste caso: INCONSISTÊNCIA, uma vez que dois alunos tem o mesmo CPF. Em um sistema de arquivos, isso poderia passar desapercebido mas em um banco de dados, existe o controle de **integridade de chave primária** que não deixa isso acontecer.
(BRAGA, 2013. p. 12)



Sistema de Gerenciamento de Banco de Dados (Estrutura)

Sistema de Gerenciamento de Banco de Dados (SGBD) em sua composição possui um conjunto de software para gerenciar um banco de dados.

Veja na figura a seguir.

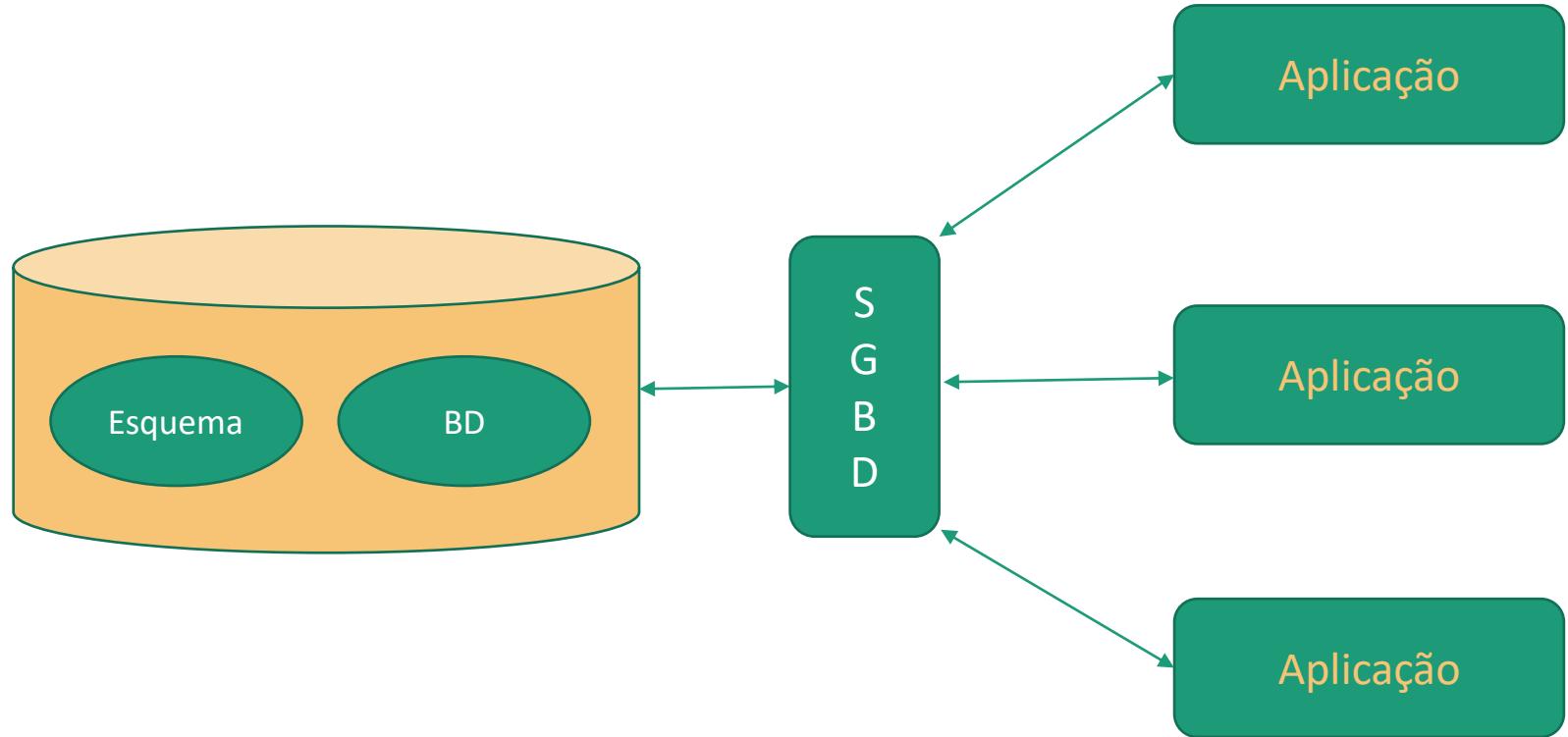


Sistema de Gerenciamento de Banco de Dados (Estrutura)

O Banco de Dados é em si um repositório de informações organizadas que se relacionam. Isso quer dizer que os dados serão armazenados nos arquivos relacionados ao SGBD e também ao esquema que será montado pelo SGBD.

O SGBD permite armazenamento e acesso multiusuário eficiente a uma grande quantidade de dados armazenados, além de garantir a integridade e segurança dos dados.

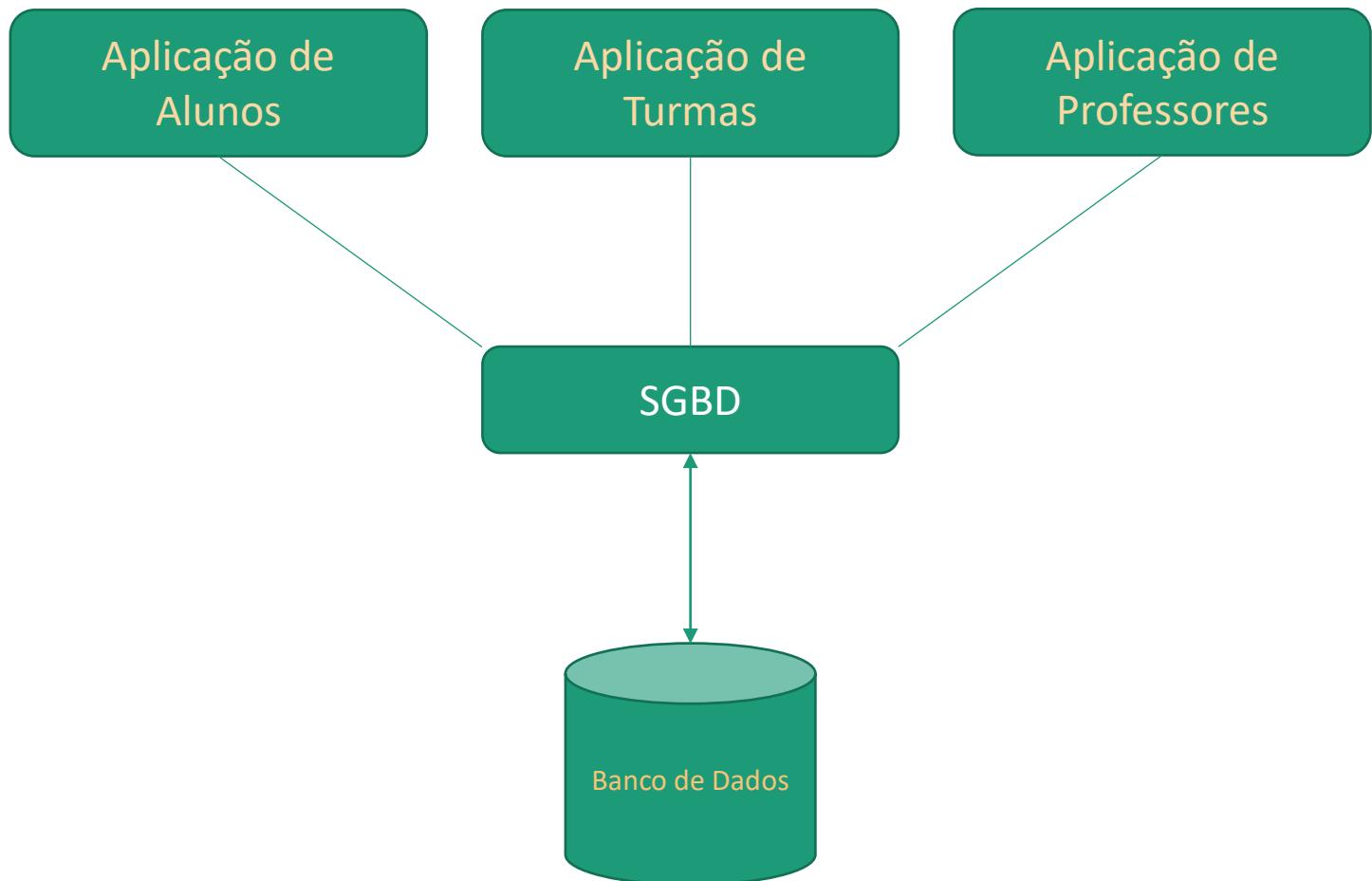
- Ao lado temos uma visão minimizada do que realmente um SGBD é em uso totalidade. Veremos mais adiante que ele é bem mais complexo do que isso.



Sistema de Gerenciamento de Banco de Dados (Estrutura)

Usando com ponto de partida uma aplicação, o manuseio de dados com o uso dos SGBDs se tornam simplificados, ou seja, as informações são integradas no Banco e o SGBD faz todo o controle de inconsistência e redundância.

Vejamos ao lado, como ficaria o exemplo dado no slide 5, quando começamos a falar sobre os problemas com base de dados em arquivos.



Sistema de Gerenciamento de Banco de Dados

Instâncias e Esquemas

Estes termos, embora relacionados, não significam a mesma coisa.

Um esquema de banco de dados é um esboço de um banco de dados planejado. Na verdade, ele não contém dados.

Uma instância de banco de dados, por outro lado, é um retrato de um banco de dados da forma como existia em um determinado momento. Sendo assim, instâncias de banco de dados podem mudar ao longo do tempo, enquanto um esquema de banco de dados é geralmente estático, já que é difícil mudar a estrutura de um banco de dados a partir do momento que estiver operacional.

Esquemas e instâncias de banco de dados podem se afetar mutuamente por meio de um sistema de gerenciamento de banco de dados (SGBD). O SGBD assegura que cada instância de banco de dados esteja em conformidade com as restrições impostas pelos designers do banco de dados no esquema de banco de dados.

→ Esquema

id	nome	numero	turma
1	Laura Dern	10	A
2	Amy Adams	5	B
3	Ana Kendrick	18	C
4	Antony Hopkins	15	D

→ Instância

Sistema de Gerenciamento de Banco de Dados

Instâncias e Esquemas

Estes termos, embora relacionados, não significam a mesma coisa.

Um esquema de banco de dados é um esboço de um banco de dados planejado. Na verdade, ele não contém dados.

Uma instância de banco de dados, por outro lado, é um retrato de um banco de dados da forma como existia em um determinado momento. Sendo assim, instâncias de banco de dados podem mudar ao longo do tempo, enquanto um esquema de banco de dados é geralmente estático, já que é difícil mudar a estrutura de um banco de dados a partir do momento que estiver operacional.

Esquemas e instâncias de banco de dados podem se afetar mutuamente por meio de um sistema de gerenciamento de banco de dados (SGBD). O SGBD assegura que cada instância de banco de dados esteja em conformidade com as restrições impostas pelos designers do banco de dados no esquema de banco de dados.

→ Esquema

id	nome	numero	turma
1	Laura Dern	10	A
2	Amy Adams	5	B
3	Ana Kendrick	18	C
4	Antony Hopkins	15	D

→ Instância

Independência de Dados

É a capacidade de fazer alterações na definição dos esquemas em determinado nível, sem afetar o esquema do nível superior.

Independência de dados física:

Faz uma modificação no esquema do BD sem que, com isso, qualquer programa aplicativo precise ser reescrito. Exemplificando essa teoria, nós poderíamos ter modificado a estrutura de dados utilizada para o armazenamento dos dados sem afetar os demais níveis. O que vocês acham disso? É claro que não vai afetar em nada os níveis acima pois a estrutura lógica ficará a mesma!

Independência de dados lógica:

Faz uma modificação no esquema do BD sem que, com isso, qualquer programa aplicativo precise ser reescrito. Um exemplo concreto seria: Vamos considerar os alunos do curso. Imaginemos que além dos dados como nome, data de nascimento, turma e disciplina, teríamos também que agora armazenar os endereços dos alunos, ou seja, teríamos que modificar a tabela alunos para incluir este novo campo! Qual o impacto disso? Com certeza um pouco maior que a mudança no nível físico, pois pelo menos na aplicação onde é feito o cadastramento dos alunos, teríamos que fazer mudanças!!!

Modelo de Dados

Descreve a estrutura conceitual dos dados armazenados no banco de dados, detalhando a organização dos dados internamente no banco.

Como exemplo, se criarmos um banco de dados para um sistema acadêmico. O modelo de dados a ser criado deverá descrever o que este banco de dados deve armazenar:

- Para **alunos**: Matrícula, nome, endereço, cidade, estado...
- Para as **notas**: Matrícula, turma, aluno...

O modo como as informações são organizadas no banco também pode mudar. Assim, um BD pode ser classificado em:

- **Relacional (nossa estudo)**
- Redes
- Hierárquico
- Orientado a objetos
- Objeto-relacional

Sistema de Gerenciamento de Banco de Dados

Modelo Relacional

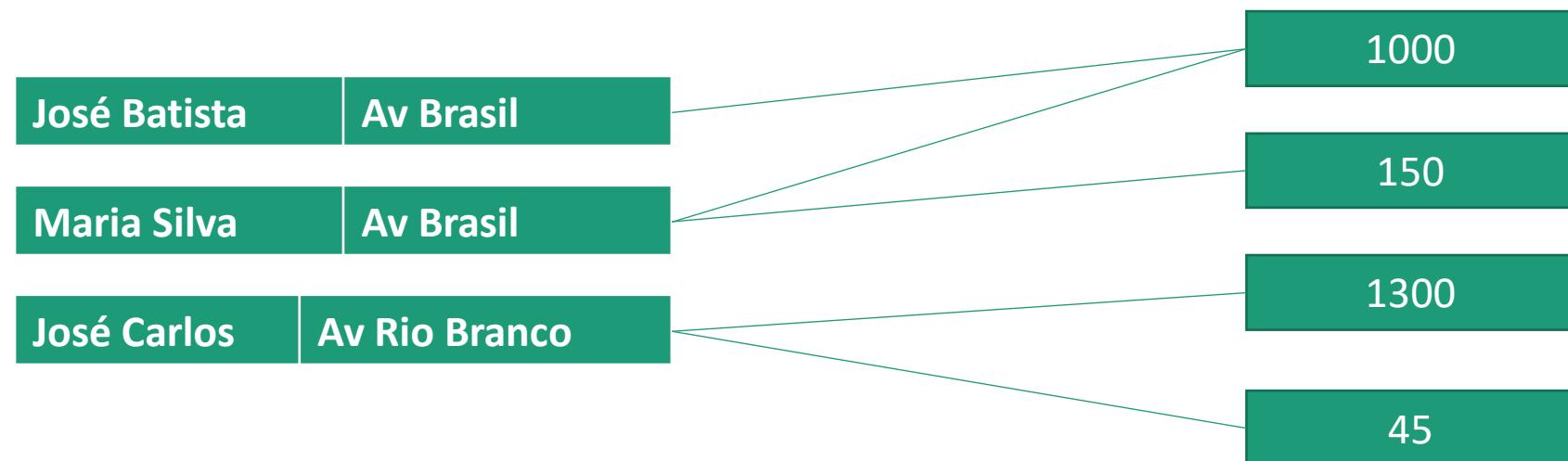
Este modelo é amplamente utilizado atualmente pelos Sistemas Gerenciadores de Banco de Dados. A **simplicidade** e a uniformidade são suas características mais relevantes em conjunto com sua linguagem padronizada para consulta (SQL - Structured Query Language).

Um conjunto de tabelas é usado para representar tanto os dados quanto a relação entre eles. As ligações entre as tabelas é feita por meio dos valores dos atributos ou colunas, conforme descrito posteriormente. Veja exemplo a seguir:

id	nome	Endereço	Cidade	Estado	Cep
1	John Doe	Rua Roque Picorelli, 15 - Grajaú	Juiz de Fora	MG	36052-330
2	Jane Silva	Rua Adofo Kirchmaier, 166 – São Pedro	Juiz de Fora	MG	36058-120
3	João Ferreira	Rua São Mateus, 1350 – São Mateus	Barbacena	MG	37055-330
4	Maria Pereira	Rua Halfeld, 72 - Centro	Juiz de Fora	MG	35100-220

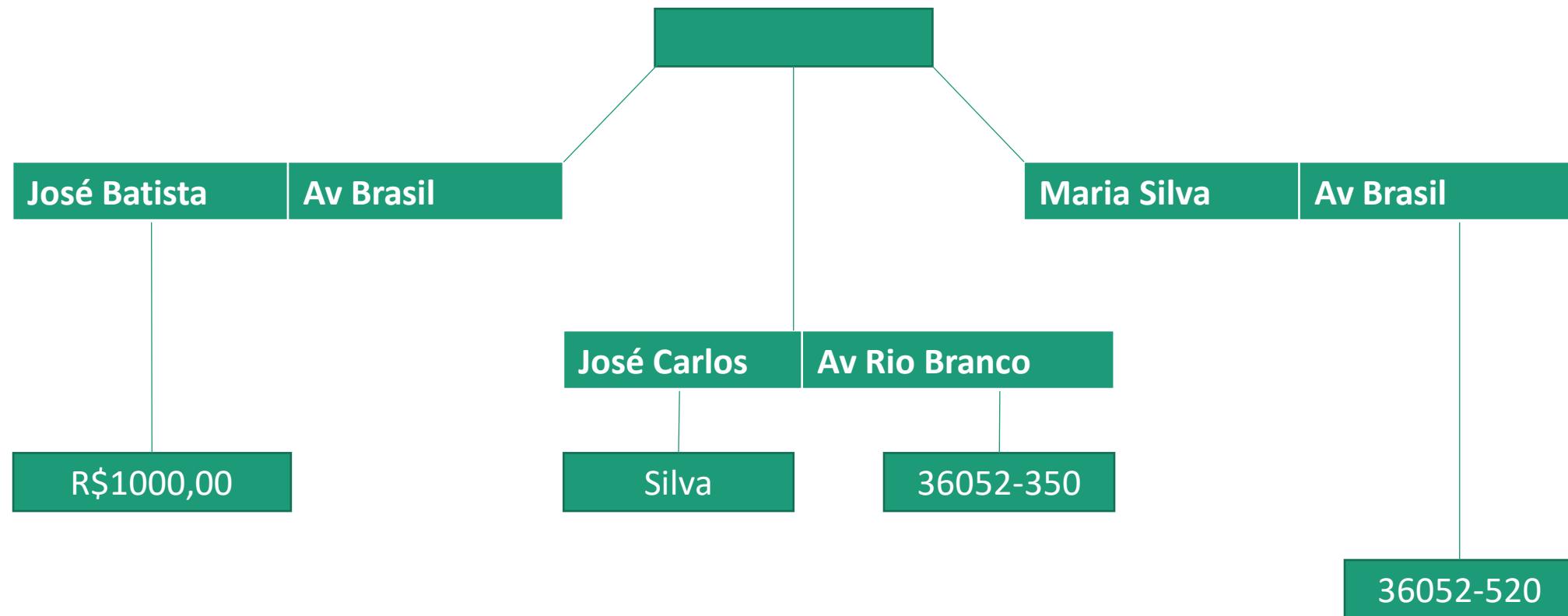
Modelo de Redes

Uma coleção de registro que são concatenados uns aos outros por meio de ligações é considerada um banco de dados em rede. Não é um modelo de Banco de Dados mais utilizado pelos SGBSs, sendo mais ligado a forma de se acessar o dado fisicamente, ao contrário do modelo relacional, que permite uma visão mais lógica do banco de dados. Os registros dos Banco de Dados são organizados por um conjunto arbitrário de grafos. Veja exemplo:



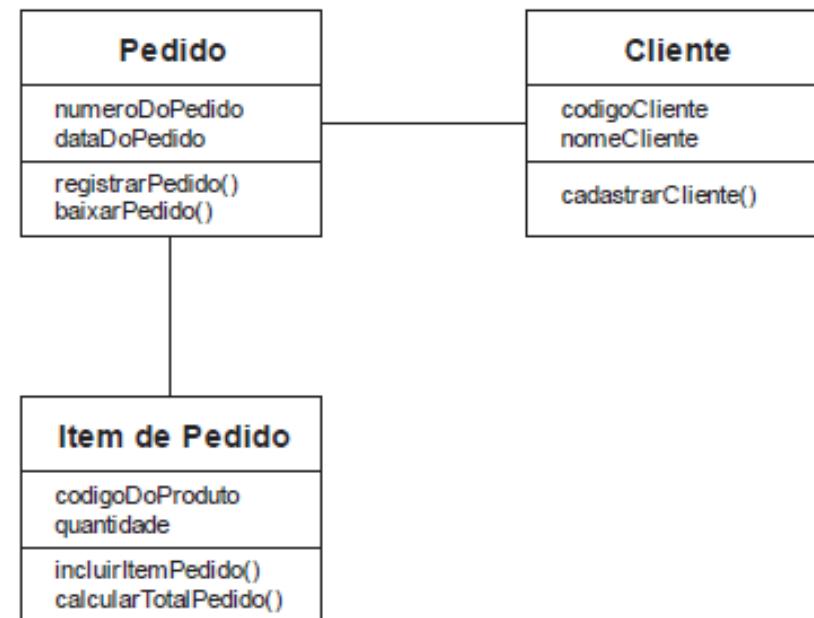
Modelo Hierárquico

Este modelo de BD se assemelha bastante com o modelo de Rede pois os dados e suas relações são representados por registros e ligações. A diferença é que no modelo hierárquico os registros estão organizados em árvores ao invés de grafos aleatórios como modelo de redes. O modelo hierárquico também não é mais utilizado como modelo de BDs atuais mesmo tendo como grande vantagem o desempenho das consultas por utilizar uma estrutura em árvore.



Modelo Orientado a Objetos

Este modelo apareceu para suprir uma lacuna no armazenamento de dados providos a partir das linguagens de programação que utilizam a Orientação a Objetos. Se levarmos em consideração as aplicações onde trabalhamos a partir de objetos (como em Java), por que armazenar os dados dos objetos em tabelas e não em objetos? É nessa premissa que se baseou a construção de BDs orientados a objetos, pois conceitos como classes, herança, métodos, associações entre classes foram incorporados nos bancos de dados. Entretanto, mesmo de toda a flexibilidade do modelo, este tipo de banco de dados não foi bem aceito no mercado.



Conceitos de Manipulação de Banco de Dados

Linguagem de Definição de Dados (DDL - Data-Definition Language)

Para manipularmos a criação de uma Base de Dados precisamos esquematizar um Banco de Dados, ou seja, criação de tabelas, índices, etc., para isso utilizamos a Linguagem de Definição de Dados DDL com os seguintes comandos:

CREATE TABLE: para criar uma tabela

ALTER TABLE: alteração de tabelas

DROP TABLE: apagar uma tabela

CREATE INDEX: criação de índices

ALTER INDEX: alteração de índices

DROP INDEX: apagar um índice

Linguagem de Manipulação de Dados (DML - Data Manipulation Language)

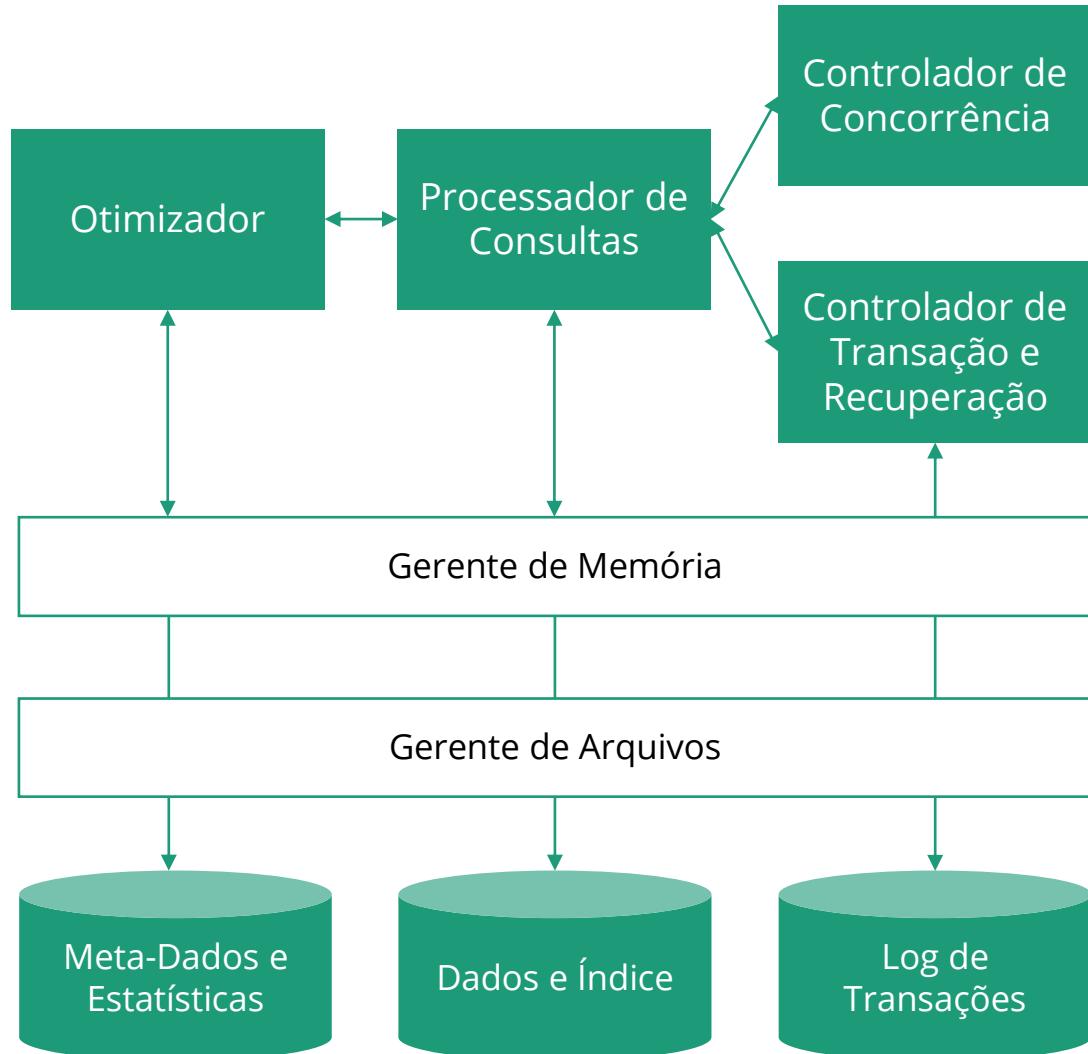
A DML permite a manipulação de dados. Podemos realizar as seguintes operações a partir da LMD:

- SELECT - recuperar dados de um banco de dados
- INSERT - inserir dados em uma tabela
- UPDATE - atualiza os dados existentes em uma tabela
- DELETE - Exclui todos os registros de uma tabela de banco de dados

A DML viabiliza o acesso (manipulação) dos dados de forma compatível ao modelo de dados apropriado. Tanto a DDL quanto a LMD fazem parte da linguagem SQL (Structured Query Language), que veremos mais adiante.

Principais componentes de um SGBD

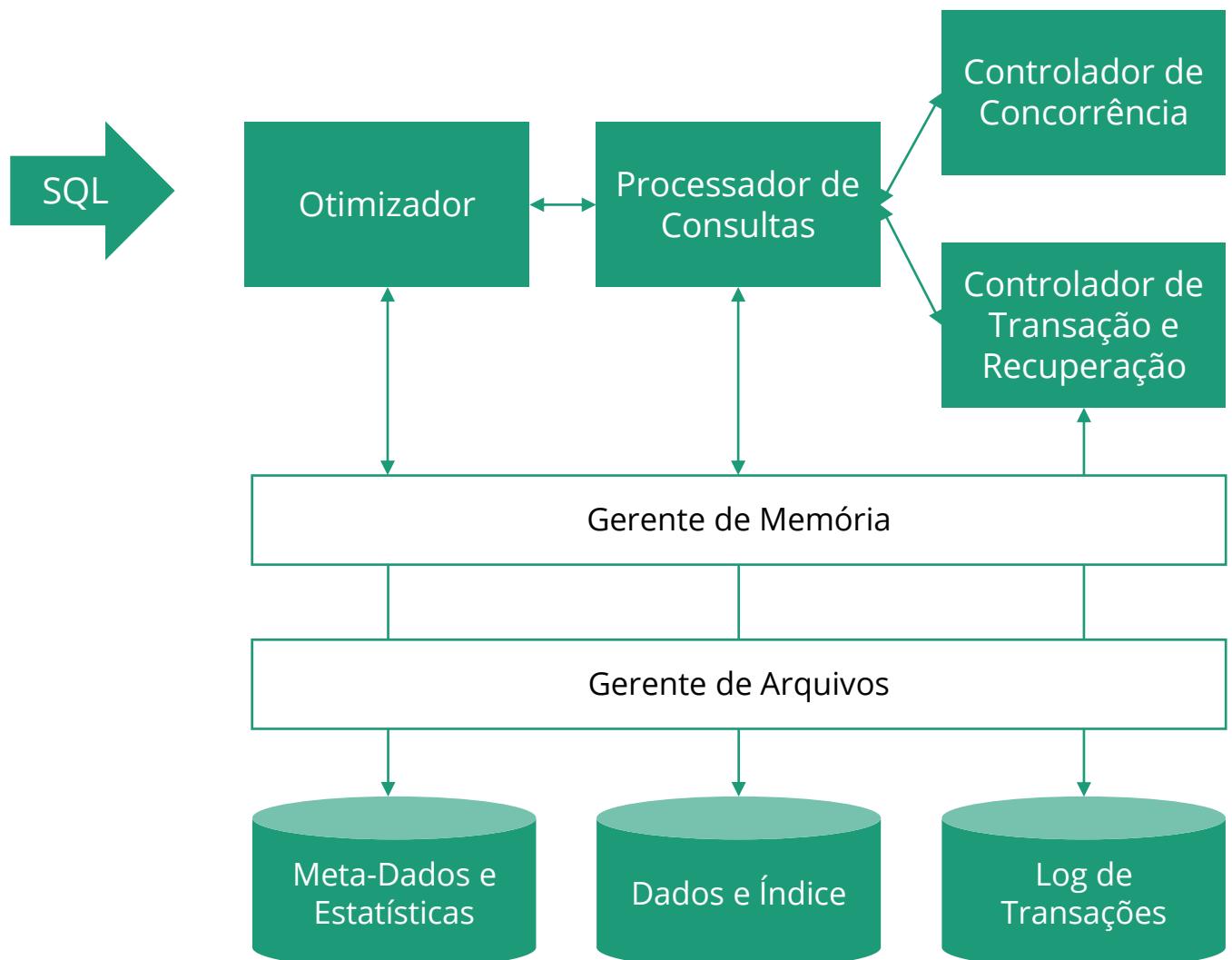
Um SGBD é dividido em módulos específicos, de modo a compreender a todas as suas funções, algumas delas advindas do sistema operacional. Os módulos podem ser organizados em dois grupos: o de processamentos de consultas e o de administração do armazenamento de dados.



Principais componentes de um SGBD

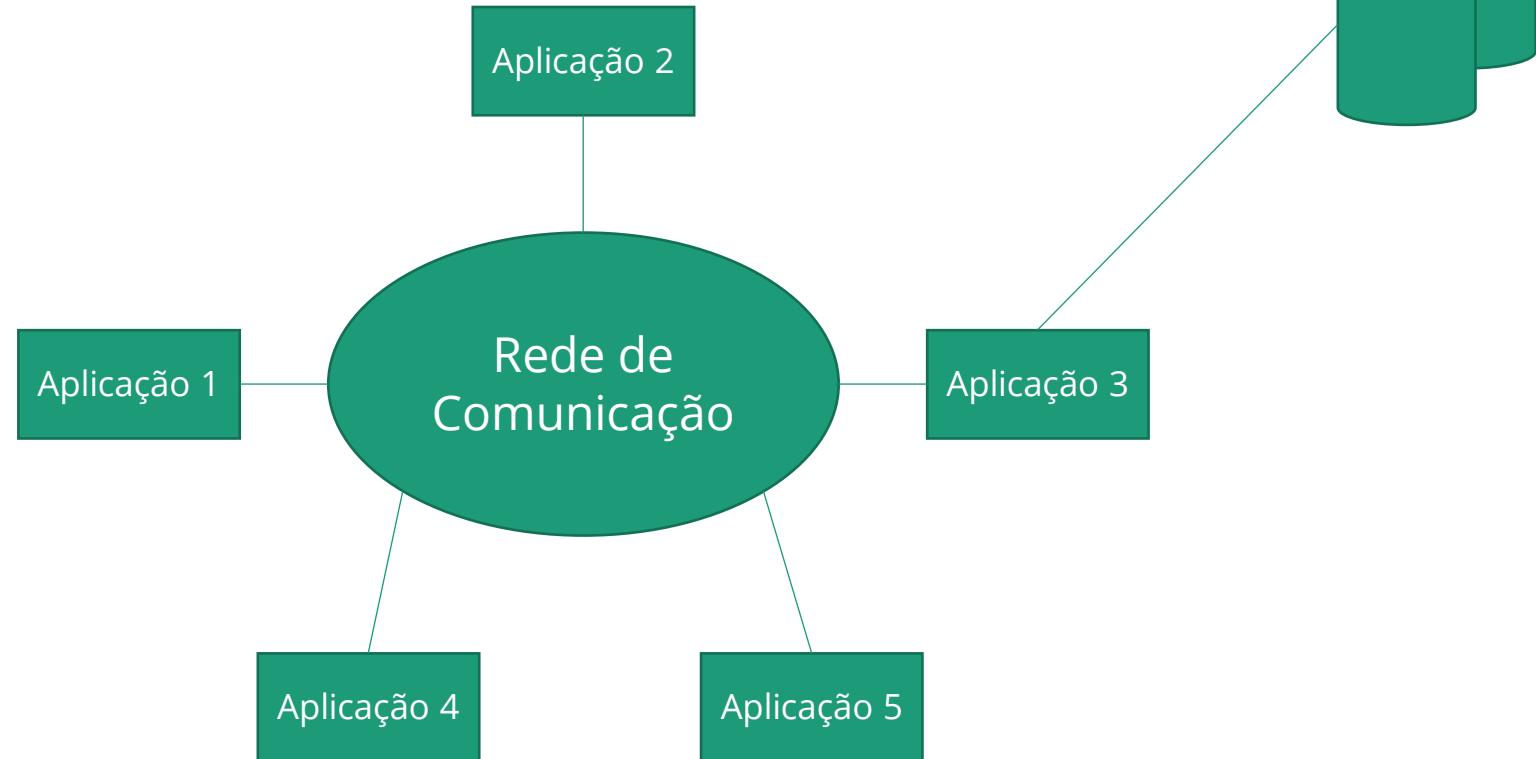
Processamento de Consultas: Constituído pelo **otimizador e processador de consultas**. Seu principal objetivo é receber uma requisição do usuário especificada em SQL e melhorar esta requisição em termos de desempenho (fazer com que a consulta seja executada o menor tempo possível) e acessar o e disponibilizar os dados requeridos pelo usuário;

Administração do armazenamento de dados: Garante o estado consistente do BD, não permitindo acesso não autorizado aos dados e nem o acesso simultâneo aos dados. Gerencia a alocação de espaço em disco e as estruturas de dados utilizadas para armazenar as informações. Outro componente importante é o gerenciamento das trocas de dados entre o disco e a memória, o que é feito pelo **gerente de memória**.



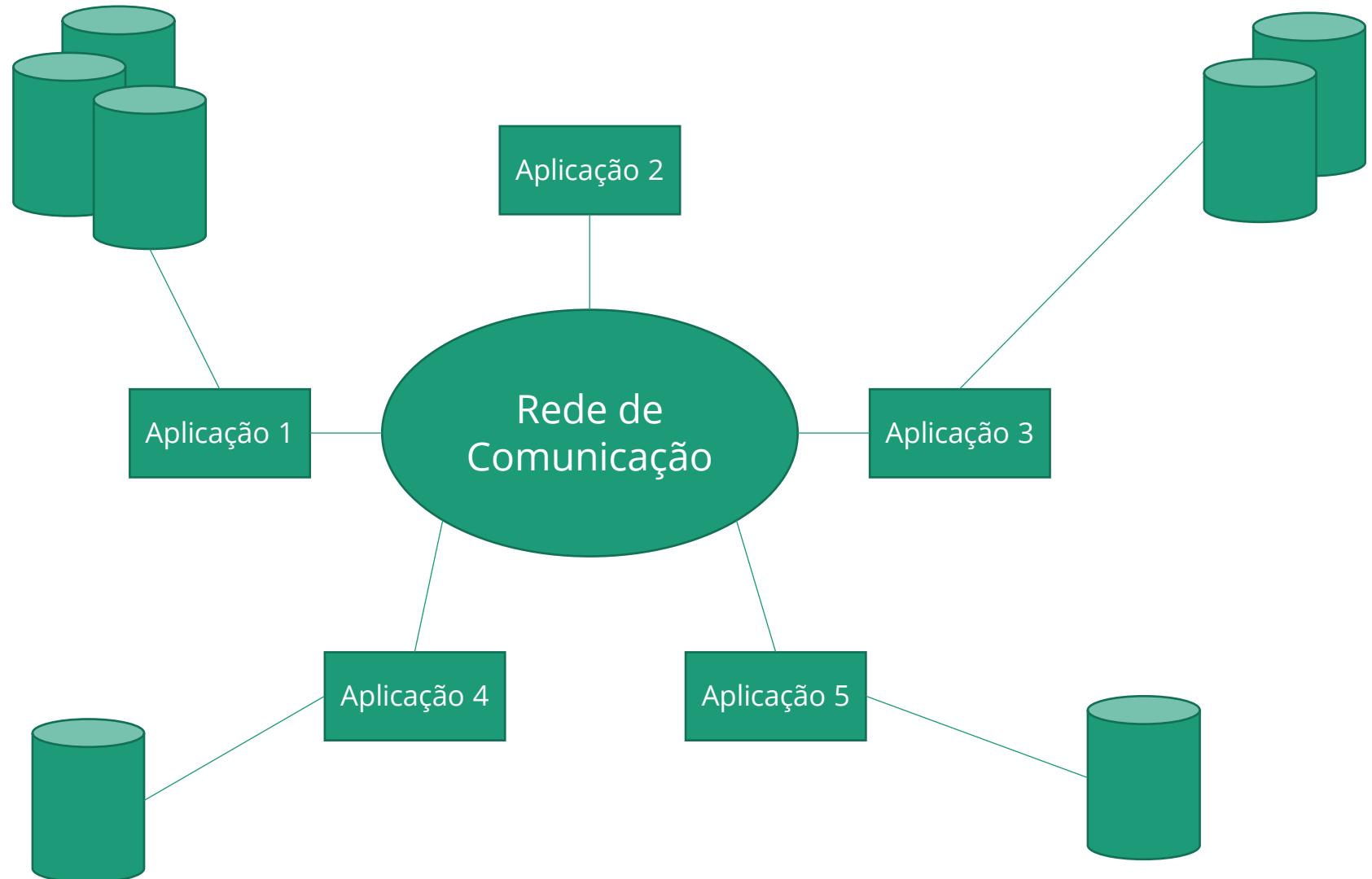
Arquitetura de um SGBD

Arquitetura Centralizada: O Sistema Gerenciador de Banco de Dados é executado e armazenado em uma máquina, podendo ser consultado através de outros computadores com acesso a este computador denominado servidor de SGBD. O acesso pode ser multiusuário, mas o SGBD está centralizado é uma máquina (o servidor).



Arquitetura de um SGBD

Arquitetura Distribuída: O Sistema Gerenciador de Banco de Dados é distribuído em várias máquinas e/ou aplicações/sites, sendo que o processamento de qualquer requisição é gerenciada de forma a acessar as "partes" onde estão armazenadas as informações relevantes.



Usuários de um banco de dados

Usuários leigos: Fazem acesso de alto nível ao BD por meio das telas de aplicativos que acessam o base de dados;

Usuários avançados: usuários que trabalham com os BDs por intermédio de telas específicas disponíveis no SGBD. Desenvolvem consultas SQL e executam sem a necessidade de escrever uma aplicação para esse fim;

Programadores aplicações: usuários com formação em computação que constroem aplicações, por meio de ferramentas (compiladores) destinadas para esse fim. Com essas ferramentas, desenvolvem interfaces para as aplicações, adicionando formulários e relatórios que acessam o bancos de dados;

Administrador de Banco de Dados (DBA – DataBase Administrator): tipo de usuário especializado. Cabe a ele a administração dos BDs, definir a melhor estrutura de armazenamento desses dados, definir aspectos de segurança, programação de copias de segurança, dentre outros.

Atividade 1 BD

1. Cite quatro sistemas que você tenha usado/visualizado recentemente e que provavelmente tenha um SGBD para realizar a persistência dos dados.
2. Detalhe quais são as principais funções de um administrador de um banco de dados
3. Descreva pelos menos 3 tabelas que você acredita que precisariam ser usadas para armazenar informações em um sistema de redes sociais como o Facebook.
4. Liste os primeiros passos que você seguiria para a criação de um banco de dados para uma empresa e justifique.
5. Defina as principais categorias de modelos de dados.
6. Dê uma vantagem e uma desvantagem de se usar um SGBD em um sistema, justifique sua resposta.

Introdução

Os bancos de dados relacionais tiveram um grande benefícios com a implementação da linguagem de consulta SQL que por sua vez fez com que ele se estabelecesse no mercado. Desde sua definição como padrão, em 1986, passou por diversas revisões, gerando publicações de novas versões. A linguagem SQL não é somente uma linguagem de consulta. Ela possui diversas outras funcionalidades, veja a lista a seguir :

Linguagem de Definição de Dados (DDL)

Agrega comandos para definição de esquemas de relações, exclusão de relações, criação de índices e modificações do esquema de relações. Alguns exemplos de comandos:

CREATE TABLE - Cria uma nova tabela com seus campos e define as restrições de campo.

ALTER TABLE - Altera as definições de campos e de restrições.

CREATE INDEX - Cria um novo índice em uma tabela existente.

DROP - Exclui uma tabela existente de um banco de dados ou exclui um índice existente de uma tabela.

Linguagem de Consulta SQL

Linguagem de manipulação de dados (DML) - agrupa comandos para inserção, exclusão e modificação de registros no banco de dados.

INSERT : inserção de registros.

DELETE : deleção de registros.

UPDATE : atualização de registros.

SELECT : seleção de registros.

Linguagem de Controle de Dados - DCL (Data Control Language) - agrupa comandos para concessão/autorização de acessos as informações no banco de dados.

GRANT : concessão de privilégios a tabelas e visões

o REVOKE : revogação de privilégios a tabelas e visões

Linguagem para Controle de Transação - possibilita o controle do processamento das transações (ações executadas no banco de dados)

COMMIT : efetiva uma alteração no banco de dados

ROLLBACK : desfaz uma alteração antes de ser efetivada no banco

SAVEPOINT : permite uma subdivisão lógica de uma transação longa

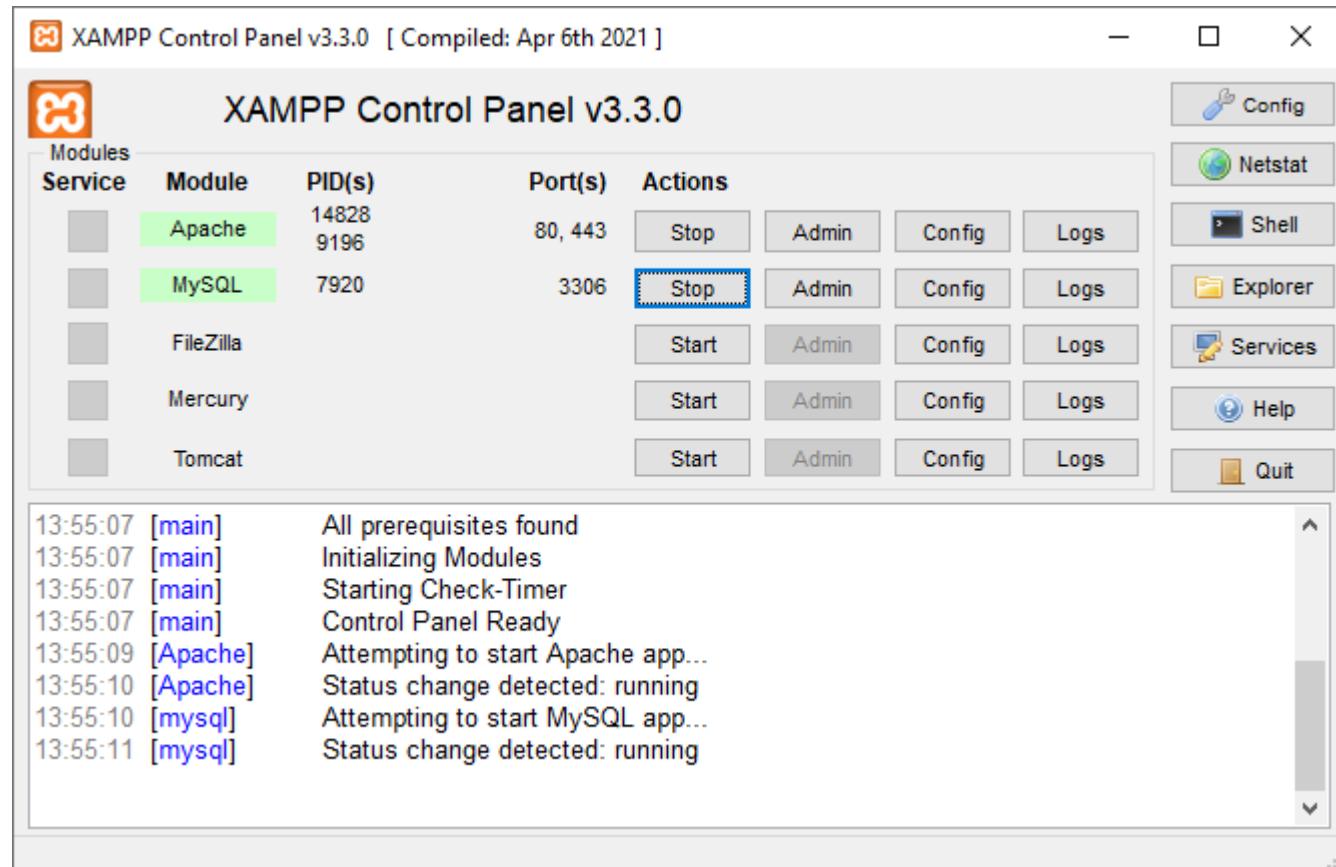
Restrições de integridade: são usados para garantir a exatidão e a consistência dos dados em uma Banco de dados relacional. Ou seja, garantir que dados representem assertivamente a realidade modelada. A integridade dos dados é tratada nas bases de dados através do conceito de integridade relacional e é garantida pelo próprio SGBD.

STORED PROCEDURES (procedimentos armazenados no banco)

TRIGGERS (gatilhos)

Criando um banco de Dados

Com XAMPP devidamente instalado siga os passo

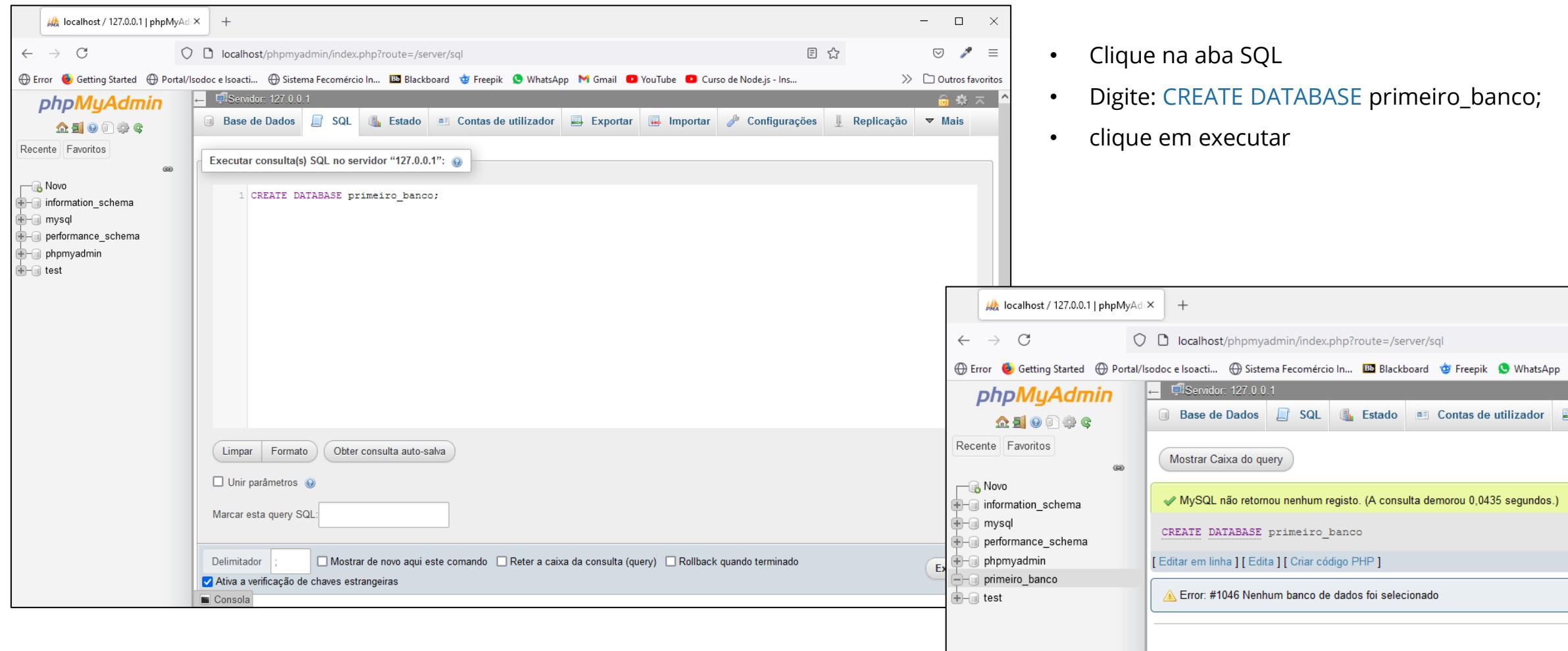


- No painel de controle do XAMPP dê start nos módulos Apache e MySQL.
- No módulo SQL clique em Admin, você será levado a página do PHPMyAdmin, veja a seguir

Linguagem de Consulta SQL

Criando um banco de Dados

Com a janela do browser aberta vá até a aba SQL



The image shows two screenshots of the phpMyAdmin interface. The left screenshot shows the 'SQL' tab selected, with the query 'CREATE DATABASE primeiro_banco;' entered into the query editor. The right screenshot shows the 'Base de Dados' tab selected, with the newly created database 'primeiro_banco' listed in the database tree.

- Clique na aba SQL
- Digite: **CREATE DATABASE primeiro_banco;**
- clique em executar

Tipos primitivos de um banco de dados

Numérico	Inteiro	TinyInt, SmallInt, Int, MediumInt, BigInt
	Real	Decimal, Float, Double, Real
	Lógico	Bit, Boolean
Data/Tempo	Date, DateTime,TimeStamp, Time, Year	
Literal	Caractere	Char, VarChar
	Texto	TinyText, Text, MediumText, LongText
	Binário	TinyBlob, Blob, MediumBlob, LongBlob
	Coleção	Enum, Set
Espacial	Geometry, Point, Polygon, MultiPolygon	

Linguagem de Consulta SQL

Criando uma tabela no banco de dados

Com a janela do browser aberta vá até a aba SQL

The screenshot shows the phpMyAdmin interface on a web browser. The URL is `localhost/phpmyadmin/index.php?route=/database/sql&db=test`. The left sidebar lists databases: Novo, information_schema, mysql, performance_schema, phpmyadmin, primeiro_banco, and test. The 'SQL' tab is selected. In the main area, there is a text input field labeled 'Executa comando(s) SQL na base de dados test:' containing the following SQL code:

```
1 /*ABRE UM BANCO DE DADOS*/
2 USE primeiro_banco;
3
4 /*CRIA UMA TABELA NO BANCO DE DADOS ABERTO*/
5 CREATE TABLE aluno(
6     matricula INT(4),
7     nome VARCHAR(256),
8     idade TINYINT(3),
9     turma CHAR(1),
10    turno CHAR(6)
11 );
```

An orange arrow points from the right side of the code in the input field towards the right panel. The right panel shows the results of the query execution:

```
1 MySQL não retornou nenhum registo. (A consulta demorou 0,0004 segundos.)
2 /*ABRE UM BANCO DE DADOS*/ USE primeiro_banco
3 [ Editar em linha ] [ Edita ] [ Criar código PHP ]
4 MySQL não retornou nenhum registo. (A consulta demorou 0,3195 segundos.)
5 /*CRIA UMA TABELA NO BANCO DE DADOS ABERTO*/ CREATE TABLE aluno( matricula INT(4), nome VARCHAR(256), idade TINYINT(3), turma CHAR(1), turno CHAR(6) )
6 [ Editar em linha ] [ Edita ] [ Criar código PHP ]
```

Linguagem de Consulta SQL

Criando uma tabela no banco de dados

No PHPMyAdmin, selecione a tabela criada e clique em estrutura.

The screenshot shows the 'Estrutura da tabela' (Table Structure) page in PHPMyAdmin. The table 'alunos' has the following structure:

#	Nome	Tipo	Agrupamento (Collation)	Atributos	Nulo	Predefinido	Comentários	Extra	Acções	
1	matricula	int(4)			Sim	NULL		Muda	Elimina	Mais
2	nome	varchar(256)	utf8mb4_general_ci		Sim	NULL		Muda	Elimina	Mais
3	idade	tinyint(3)			Sim	NULL		Muda	Elimina	Mais
4	turma	char(1)	utf8mb4_general_ci		Sim	NULL		Muda	Elimina	Mais
5	turno	char(6)	utf8mb4_general_ci		Sim	NULL		Muda	Elimina	Mais

Below the table structure, there are several buttons and links for managing the table:

- Back arrow, Select all checkboxes, 'Com os selecionados:' dropdown, 'Procurar' button, 'Muda' button, 'Elimina' button, 'Primária' button, 'Único' button, 'Índice' button.
- 'Espacial', 'Texto Completo', 'Adicionar à(s) coluna(s) central(is)', 'Remover da(s) coluna(s) central(is)' buttons.
- 'Imprimir', 'Propor uma estrutura de tabela', 'Acompanhar tabela', 'Mover campo(s)', 'Normalizar' buttons.
- 'Adicionar' button with a dropdown menu showing '1 campo(s) após turno', and an 'Executar' button.
- 'Índices' button.

Linguagem de Consulta SQL

Criando uma tabela no banco de dados

Também podemos ver a estrutura da tabela utilizando o comando DESCRIBE seguido do nome da tabela

The screenshot shows a MySQL query interface with the following details:

Mostrar Caixa do query

Aviso: A seleção atual não contém uma coluna exclusiva. Os recursos de edição de grade, caixa de seleção, Editar, Copiar e Apagar não estão disponíveis.

Mensagem: A sua consulta SQL foi executada com êxito.

Consulta SQL:

```
1 DESCRIBE aluno;
```

Opções:

[Editar em linha] [Edita] [Criar código PHP]

Opções de resultado:

+ Opções

Field	Type	Null	Key	Default	Extra
matricula	int(4)	YES		NULL	
nome	varchar(256)	YES		NULL	
idade	tinyint(3)	YES		NULL	
turma	char(1)	YES		NULL	
turno	char(6)	YES		NULL	

Operações resultantes das consultas

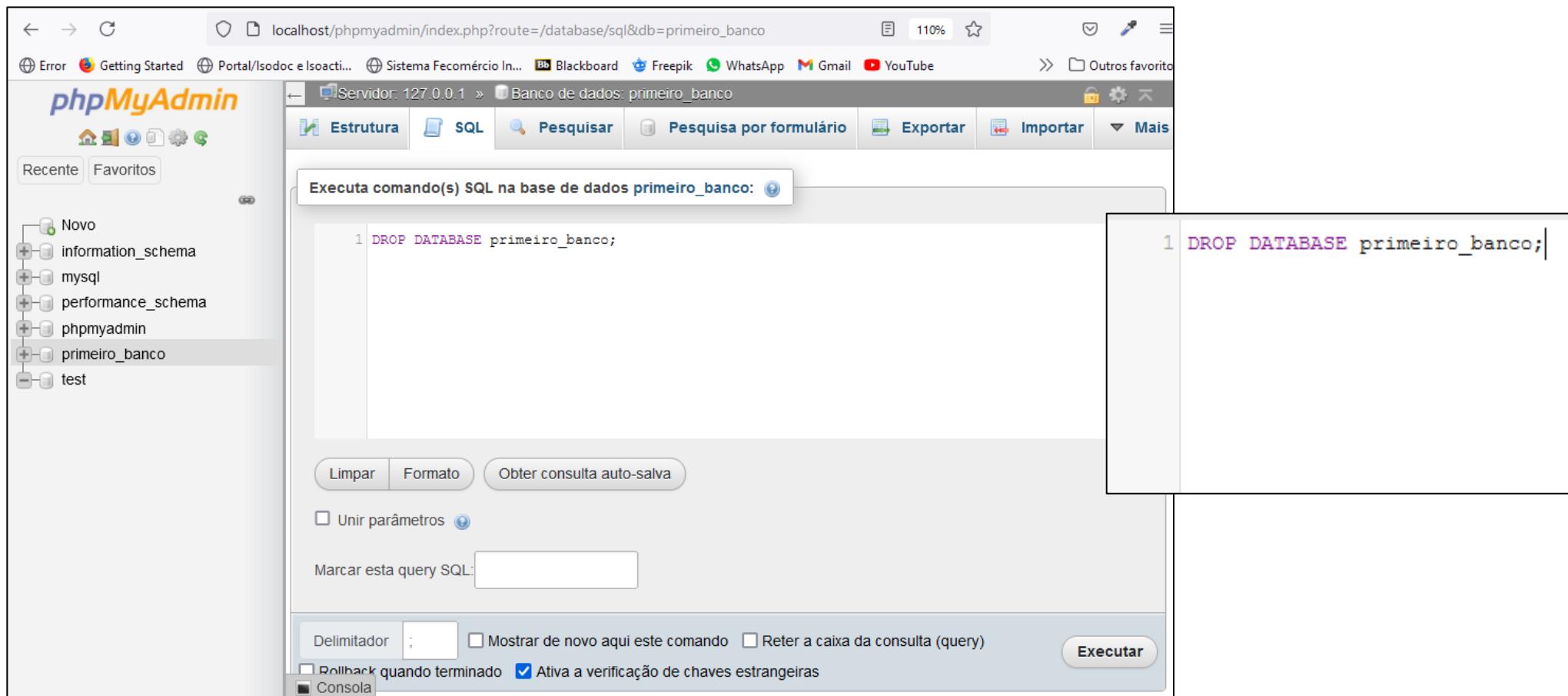
[Imprimir](#) [Copiar para área de transferência](#) [Criar visualização](#)

Linguagem de Consulta SQL

Apagando um banco de Dados

Para apagar um banco de dados basta utilizar o comando Drop Database.

Tenha **cuidado** antes de eliminar um banco de dados. A exclusão de um banco de dados resultará na perda de informações completas armazenadas no banco de dados!



Linguagem de Consulta SQL

Utilizando Constraints para resolver problemas de acentuação

Quando criamos nosso banco de dados é possível logo de início configurar a codificação do país para termos nossos caracteres apresentados da forma correta. O ponto e vírgula na última linha indica o fim do comando.

The screenshot shows the MySQL Workbench interface. In the top navigation bar, the 'SQL' tab is selected. Below it, a query editor window displays the following SQL code:

```
1 /*RESOLVENDO PROBLEMA DE ACENTUAÇÃO JÁ NA CRIAÇÃO DO BD*/
2 CREATE DATABASE aluno
3 DEFAULT CHARACTER SET utf8
4 DEFAULT COLLATE utf8_general_ci;
```

Below the code, a message box indicates: "MySQL não retornou nenhum registo. (A consulta demorou 0,0043 segundos.)". The bottom status bar shows the command was executed successfully: "CREATE DATABASE primeiro_banco DEFAULT CHARACTER SET utf8 DEFAULT COLLATE utf8_general_ci [Editar em lir]".

Melhorando a criação de tabelas

Vamos melhorar a nossa estrutura de tabela incluindo tipos primitivos mais consistentes como date, enum e decimal.

Chaves primárias (em inglês, Primary keys ou "PK"): sob o ponto de vista de um banco de dados relacional, referem-se aos conjuntos de um ou mais campos, cujos valores, considerando a combinação de valores em caso de mais de uma chave primária, nunca se repetem na mesma tabela e, desta forma, podem ser usadas como um índice de referência para criar relacionamentos com as demais tabelas do banco de dados (daí vem o nome banco de dados relacional). Portanto, uma chave primária nunca pode ter valor nulo, nem repetição.

Simplificando, quando a chave primária é simples, ou seja, é formada por um único campo da tabela, esse campo não pode ter dois ou mais registros de mesmo valor e também não pode conter nenhum registro nulo. Se a chave primária é composta, ou seja, formada por mais de um campo, os valores de cada campo podem se repetir, mas nunca a combinação desses valores. Exemplo: a tabela 'Livros_Autores' tem como chave primária (cod_livro, cod_autor). Podem existir nessa tabela os registros:

Linguagem de Consulta SQL

Tipos melhorados e chave primária

The screenshot shows the phpMyAdmin interface for a database named 'primeiro_banco'. In the left sidebar, the 'primeiro_banco' schema is selected. The main area displays a SQL query for creating a table:

```
1 CREATE TABLE aluno(
2     id INT NOT NULL AUTO_INCREMENT,
3     matricula INT(4),
4     nome VARCHAR(50) NOT NULL,
5     data_nascimento DATE,
6     turma CHAR(1) NOT NULL,
7     turno CHAR(6) NOT NULL,
8     altura DECIMAL(3,2),
9     monitor ENUM('S', 'N'),
10    cidade VARCHAR(45) DEFAULT 'Juiz de Fora',
11    PRIMARY KEY(id)
12 )DEFAULT CHARSET = utf8;
```

The query is executed successfully, as indicated by the message: "MySQL não retornou nenhum registo. (A consulta demorou 0,1426 segundos.)". The generated SQL code is also shown in the results panel.

Linguagem de Consulta SQL

Usando Describe para ver a estrutura

The screenshot shows the MySQL Workbench interface with the 'Estrutura' (Structure) tab selected. At the top, there are several tabs: Estrutura, SQL, Pesquisar, Pesquisa por formulário, Exportar, Importar, Operações, and Mais. Below the tabs, there are buttons for [Editar em linha], [Edita], and [Criar código PHP].

A table structure is displayed with the following columns:

Field	Type	Null	Key	Default	Extra
id	int(11)	NO	PRI	NULL	auto_increment
matricula	int(4)	YES		NULL	
nome	varchar(50)	NO		NULL	
data_nascimento	date	YES		NULL	
turma	char(1)	NO		NULL	
turno	char(6)	NO		NULL	
altura	decimal(3,2)	YES		NULL	
monitor	enum('S','N')	YES		NULL	
cidade	varchar(45)	YES		Juiz de Fora	

Below the table, there is a section titled 'Operações resultantes das consultas' (Operations resulting from queries) containing buttons for Imprimir (Print), Copiar para área de transferência (Copy to clipboard), and Criar visualização (Create visualization).

At the bottom, there is a 'Marcar este comando SQL' (Mark this command SQL) button, a 'Rótulo:' (Label:) input field, and a checkbox for 'Deixar todos os utilizadores acederem a este marcador' (Allow all users to access this marker). A cursor is hovering over the 'Marcar este comando SQL' button.

Linguagem de Consulta SQL

Inserindo dados na tabela do banco de dados, siga os passos:

The screenshot shows the phpMyAdmin interface for a MySQL database named 'primeiro_banco'. The current table is 'aluno'. The SQL query window contains the following code:

```
1 INSERT INTO aluno (matricula, nome, data_nascimento, turma, turno, altura, monitor, cidade)
2 VALUES (1001, 'José da Silva Caetano', '1970-05-10', 'A', 'Noite', 1.77, 'N', 'Juiz de Fora');
```

The results pane below shows the inserted data:

id	matricula	nome	data_nascimento	turma	turno	altura	monitor	cidade
1	1001	José da Silva Caetano	1970-05-10	A	Noite	1.77	N	Juiz de Fora

At the bottom of the interface, there are several configuration options and a 'Executar' (Execute) button.

Linguagem de Consulta SQL

Visualizando os dados: `SELECT * FROM tabela;`

Mostrar Caixa do query

✓ A mostrar registos de 0 - 0 (1 total, A consulta demorou 0,0011 segundos.)

```
SELECT * FROM aluno
```

Perfil [Editar em linha] [Edita] [Explicar SQL] [Criar código PHP] [Actualizar]

Mostrar tudo | Número de registos: 25 ▾ Filtrar registos: Pesquisar esta tabela

+ Opções

		id	matricula	nome	data_nascimento	turma	turno	altura	monitor	cidade		
<input type="checkbox"/>	 Edita	 Copiar	 Apagar	1	1001	José da Silva Caetano	1970-05-10	A	Noite	1.77	N	Juiz de Fora

Marcar todos Com os seleccionados:  Edita  Copiar  Apagar  Exportar

Mostrar tudo | Número de registos: 25 ▾ Filtrar registos: Pesquisar esta tabela

Operações resultantes das consultas

 Imprimir  Copiar para área de transferência  Exportar  Mostrar gráfico  Criar visualização

Linguagem de Consulta SQL

Atividade 2 Banco de Dados

Crie uma banco de dados para uma escola utilizando todos os dados da tabela abaixo, fiquem a vontade para escolher os nomes:

Num	S	Data Enturmação	Código	Nome	Sexo	Cond. Matrícula	Data Nascimento	Etnia	Bolsa Família	Transporte Escolar	Tipo de Transporte Escolar
01		28/04/2021	140013		M	NOVATO	05/03/2009	Branca	Não	Não	-
02		05/05/2021	210014		M	NOVATO	25/09/2009	Parda	Não	Não	-
03		28/04/2021	140005		M	NOVATO	26/08/2009	Branca	Não	Não	-
04		28/04/2021	120006		M	NOVATO	06/06/2007	Branca	Não	Não	-
05		28/04/2021	140014		M	NOVATO	29/06/2009	Não declarada	Sim	Sim (Rural)	Ônibus
06		28/04/2021	140011		M	NOVATO	17/07/2009	Parda	Não	Não	-
07		28/04/2021	140008		M	NOVATO	20/08/2009	Branca	Não	Não	-
08		05/05/2021	210016		M	NOVATO	22/06/2010	Não declarada	Não	Não	-
09		28/04/2021	140019		M	NOVATO	07/02/2010	Não declarada	Não	Não	-
10		05/05/2021	210018		M	NOVATO	05/10/2009	Não declarada	Sim	Sim (Rural)	Vans/Kombis
11		28/04/2021	140012		F	NOVATO	07/08/2009	Não declarada	Não	Não	-
12		04/05/2021	210013		F	NOVATO	02/10/2009	Não declarada	Não	Sim (Rural)	Micro-ônibus
13		28/04/2021	140007		F	NOVATO	28/05/2010	Parda	Não	Não	-
14		10/05/2021	210020		F	NOVATO	30/10/2009	Parda	Sim	Sim (Rural)	Vans/Kombis
15		05/05/2021	210017		F	NOVATO	01/12/2009	Branca	Sim	Não	-
16		28/04/2021	130006		F	NOVATO	03/09/2008	Branca	Sim	Não	-
17		28/04/2021	130028		F	NOVATO	09/11/2007	Parda	Não	Não	-

Referências

Braga, Regina M. M. **Fundamentos de Banco de dados**. Universidade Federal de Juiz de Fora. 2012.



Siga o Senac em Minas nas Redes Sociais:

