****

**Serviço Nacional de Aprendizagem Industrial**

**SENAI “Gaspar Ricardo Junior”**

Curso

TÉCNICO EM DESENVOLVIMENTO

DE SISTEMAS

Tema do Trabalho

Nome do autor ou autores

Cidade

Mês – Ano



**Serviço Nacional de Aprendizagem Industrial**

**SENAI “Gaspar Ricardo Junior”**

Nome do autor ou autores

Tema do Trabalho

Breve descrição do Trabalho (do que se trata)

Prof. – Orientador

Cidade

Mês – Ano

HISTÓRICO DE VERSÕES

|  |  |  |  |
| --- | --- | --- | --- |
| **Data** | **Versão** | **Descrição** | **Autor** |
| 17/11/2002 | 3.0 | Fechamento do escopo com definição de todos os requisitos a serem implementados no TG de Carlos. | Carlos R. S. Júnior |
| 12/08/2002 | 2.9 | Reestruturação do documento. Avaliação dos requisitos anteriores e criação de novos visando adequação às novas funcionalidades desejadas. | Carlos R. S. Júnior |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Sumário**

[RESUMO 3](#_Toc182377754)

[OBJETIVO 5](#_Toc182377755)

[O objetivo desta pesquisa é explorar os fundamentos teóricos das SQL Views, suas diferenças em relação às tabelas comuns, e os diferentes tipos de views disponíveis. A pesquisa abrange desde a definição básica até exemplos práticos de uso. 5](#_Toc182377756)

[INTRODUÇÃO 6](#_Toc182377757)

[1.1. Definição de SQL Views e sua função em bancos de dados 6](#_Toc182377758)

[1.2. Importância das views em sistemas de banco de dados relacionais 6](#_Toc182377759)

[Simplificam consultas complexas: Permitem que consultas complexas sejam encapsuladas em uma única view, facilitando o acesso aos dados. 6](#_Toc182377760)

[Melhoram a segurança:Podem restringir o acesso a colunas específicas de uma tabela, protegendo dados sensíveis. 6](#_Toc182377761)

[Facilitam a manutenção: Alterações na lógica de consulta podem ser feitas na view sem necessidade de modificar as aplicações que a utilizam. 6](#_Toc182377762)

[Uma das características principais do modelo é a possibilidade de relacionar várias tabelas, evitando a redundância no armazenamento de dados. O relacionamento nesse modelo pode ocorrer de três formas: 6](#_Toc182377763)

[A) Um-para-um; 6](#_Toc182377764)

[B) Um-para-muitos; 6](#_Toc182377765)

[C) Muitos-para-muitos 6](#_Toc182377766)

[2. FUNDAMENTOS TEÓRICOS DAS SQL VIEWS 7](#_Toc182377767)

[2.1. O que são views e como elas funcionam no SQL 7](#_Toc182377768)

[2.2. Diferença entre views e tabelas comuns 7](#_Toc182377769)

[Views: Não armazenam dados fisicamente, apenas a consulta que gera os dados. 7](#_Toc182377770)

[Tabelas comuns: Armazenam dados fisicamente no banco de dados. 7](#_Toc182377771)

[2.3. Tipos de views 7](#_Toc182377772)

[Simples: Baseadas em uma única tabela e sem funções agregadas. Sendo úteis para oferecer uma visão resumida ou segmentada dos dados originais. 7](#_Toc182377773)

[Complexas: Envolvem junções entre múltiplas tabelas e podem incluir funções agregadas. São especialmente úteis para simplificar consultas complexas e fornecer uma visão abrangente dos dados. 8](#_Toc182377774)

[Materializadas: Armazenam fisicamente os dados resultantes da consulta, melhorando o desempenho em consultas complexas, mas requerem atualização manual ou automática. 8](#_Toc182377775)

[3. VANTAGENS E DESVANTAGENS DE USAR VIEWS 8](#_Toc182377776)

[3.1. Vantagens 9](#_Toc182377777)

[Simplificação de consultas complexas: Views permitem encapsular consultas complexas, tornando-as mais fáceis de reutilizar e entender. 9](#_Toc182377778)

[Aumento da Segurança: Ao criar views, você pode restringir o acesso a dados sensíveis, exibindo apenas as colunas e linhas necessárias para determinados usuários. 9](#_Toc182377779)

[Facilidade de Manutenção: Views ajudam a centralizar e padronizar consultas frequentemente usadas, facilitando a manutenção e atualização. 9](#_Toc182377780)

[3.2. Desvantagens: 9](#_Toc182377781)

[Impactos de Desempenho: Quando forem baseadas em consultas complexas ou se não forem otimizadas, podem introduzir sobrecarga de desempenho. 9](#_Toc182377782)

[Limitações em Operações de Atualização: As que envolvem múltiplas tabelas ou agregações, podem não ser atualizáveis diretamente. 9](#_Toc182377783)

[Manutenção de Views Materializadas: Precisam ser atualizadas de tempos em tempos para refletir mudanças nos dados subjacentes, podendo adicionar complexidade e custo de manutenção. 9](#_Toc182377784)

[CONCLUSÃO 11](#_Toc182377785)

[BIBLIOGRAFIA 13](#_Toc182377786)

[LISTA DE FIGURAS 14](#_Toc182377787)

# 

Tema do Trabalho

# RESUMO

O vídeo fornece uma maneira poderosa de ajudá-lo a provar seu argumento. Ao clicar em Vídeo Online, você pode colar o código de inserção do vídeo que deseja adicionar. Você também pode digitar uma palavra-chave para pesquisar online o vídeo mais adequado ao seu documento.

Para dar ao documento uma aparência profissional, o Word fornece designs de cabeçalho, rodapé, folha de rosto e caixa de texto que se complementam entre si. Por exemplo, você pode adicionar uma folha de rosto, um cabeçalho e uma barra lateral correspondentes. Clique em Inserir e escolha os elementos desejados nas diferentes galerias.

Temas e estilos também ajudam a manter seu documento coordenado. Quando você clica em Design e escolhe um novo tema, as imagens, gráficos e elementos gráficos SmartArt são alterados para corresponder ao novo tema. Quando você aplica estilos, os títulos são alterados para coincidir com o novo tema.

Economize tempo no Word com novos botões que são mostrados no local em que você precisa deles. Para alterar a maneira como uma imagem se ajusta ao seu documento, clique nela e um botão de opções de layout será exibido ao lado. Ao trabalhar em uma tabela, clique no local onde deseja adicionar uma linha ou uma coluna e clique no sinal de adição.

# OBJETIVO

# O objetivo desta pesquisa é explorar os fundamentos teóricos das SQL Views, suas diferenças em relação às tabelas comuns, e os diferentes tipos de views disponíveis. A pesquisa abrange desde a definição básica até exemplos práticos de uso.

# INTRODUÇÃO

## Definição de SQL Views e sua função em bancos de dados

Uma visão (ou "view"), é uma consulta (instrução SQL) armazenada no banco de dados geralmente chamada de "tabela virtual". São usadas para simplificar consultas complexas, melhorar a segurança ao restringir o acesso a dados sensíveis e facilitar a reutilização de consultas frequentes

Gráfico, Diagrama

Descrição gerada automaticamente

Imagem1

## Importância das views em sistemas de banco de dados relacionais

## Simplificam consultas complexas: Permitem que consultas complexas sejam encapsuladas em uma única view, facilitando o acesso aos dados.

## Melhoram a segurança: Podem restringir o acesso a colunas específicas de uma tabela, protegendo dados sensíveis.

## Facilitam a manutenção: Alterações na lógica de consulta podem ser feitas na view sem necessidade de modificar as aplicações que a utilizam.

# Uma das características principais do modelo é a possibilidade de relacionar várias tabelas, evitando a redundância no armazenamento de dados. O relacionamento nesse modelo pode ocorrer de três formas:

# A) Um-para-um;

# B) Um-para-muitos;

# C) Muitos-para-muitos

# FUNDAMENTOS TEÓRICOS DAS SQL VIEWS

## O que são views e como elas funcionam no SQL

Uma view são tabelas virtuais (justamente por não fazerem parte do esquema físico da base) que é formado por declarações SELECTs, aonde retornam uma visualização de dados específica de uma ou mais tabelas de um banco de dados, permitindo que os usuários acessem dados de forma simplificada e organizada.

## Diferença entre views e tabelas comuns

# Views: Não armazenam dados fisicamente, apenas a consulta que gera os dados.

# Tabelas comuns: Armazenam dados fisicamente no banco de dados.

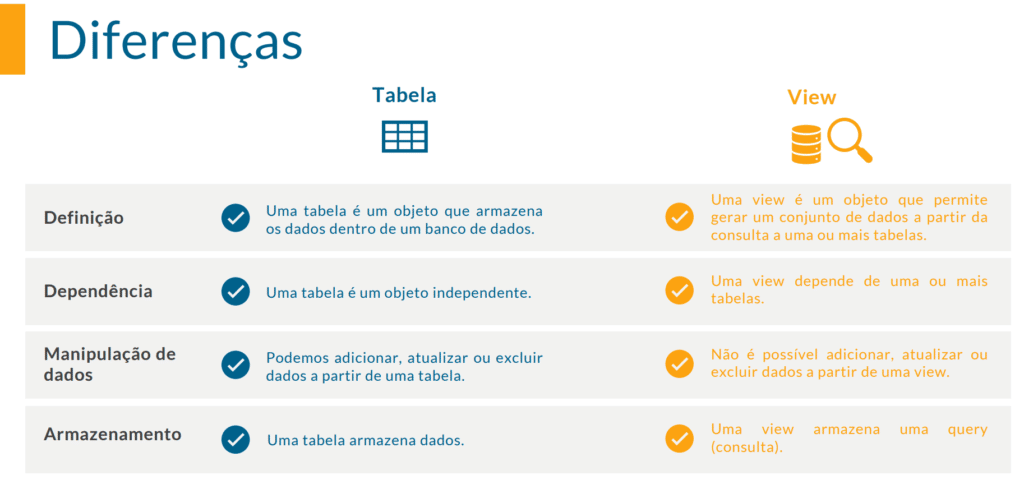


Imagem 2.

## Tipos de views

# Simples: Baseadas em uma única tabela e sem funções agregadas. Sendo úteis para oferecer uma visão resumida ou segmentada dos dados originais.

Texto

Descrição gerada automaticamente

Imagem 3.

# Complexas: Envolvem junções entre múltiplas tabelas e podem incluir funções agregadas. São especialmente úteis para simplificar consultas complexas e fornecer uma visão abrangente dos dados.

Texto

Descrição gerada automaticamente

Imagem 4.

# Materializadas: Armazenam fisicamente os dados resultantes da consulta, melhorando o desempenho em consultas complexas, mas requerem atualização manual ou automática.

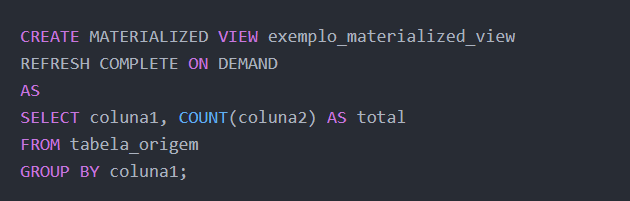


Imagem 5.

# VANTAGENS E DESVANTAGENS DE USAR VIEWS

## Vantagens

## Simplificação de consultas complexas: Views permitem encapsular consultas complexas, tornando-as mais fáceis de reutilizar e entender.

## Aumento da Segurança: Ao criar views, você pode restringir o acesso a dados sensíveis, exibindo apenas as colunas e linhas necessárias para determinados usuários.

# 

# Facilidade de Manutenção: Views ajudam a centralizar e padronizar consultas frequentemente usadas, facilitando a manutenção e atualização.

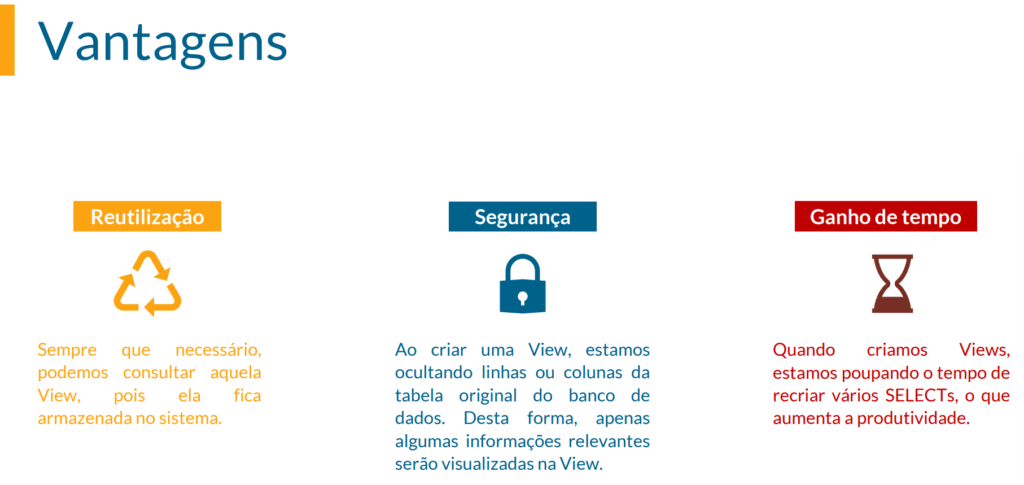


Imagem 6.

## Desvantagens:

# Impactos de Desempenho: Quando forem baseadas em consultas complexas ou se não forem otimizadas, podem introduzir sobrecarga de desempenho.

# Limitações em Operações de Atualização: As que envolvem múltiplas tabelas ou agregações, podem não ser atualizáveis diretamente.

# Manutenção de Views Materializadas: Precisam ser atualizadas de tempos em tempos para refletir mudanças nos dados subjacentes, podendo adicionar complexidade e custo de manutenção.

# PROCESSO DE CRIAÇÃO DE VIEWS NO SQL

## Instrução CREATE VIEW: sintaxe e parâmetros

A instrução CREATE VIEW é usada para criar uma view, que é uma tabela virtual baseada no resultado de uma consulta SQL. A sintaxe básica é:

Exemplo simples:

Texto

Descrição gerada automaticamente

Imagem 7

## Filtragem: seleção de colunas e linhas específicas

Uma view que seleciona colunas e linhas específicas:

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Imagem 8

## Agregação: uso de funções como SUM, AVG, COUNT

Uma view que usa funções de agregação como SUM, AVG, COUNT:

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Imagem 9

## Junção: combinar dados de várias tabelas para gerar uma nova visão

Uma view que combina dados de várias tabelas:

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Imagem 10

Exemplo complexa:

Interface gráfica do usuário, Texto

Descrição gerada automaticamente com confiança média

Imagem 11

# VIEWS ATUALIZÁVEIS E NÃO ATUALIZÁVEIS

## Possibilidade de atualizar dados diretamente em views

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

Imagem 12

## Condições para que uma view seja atualizável

- A visualização é deleitável.

- A coluna resolve para uma coluna de uma tabela (não utilizar uma operação de referência) e a opção READ ONLY não é especificado.

- Todas as colunas correspondentes dos operandos de um UNION ALL têm exatamente correspondência de tipos de dados (incluindo comprimento ou precisão e escala) e correspondência de valores padrão se a seleção completa da visualização incluir um UNION ALL.

## Exemplo prático de uma view atualizável e outra não atualizável

**Atualizável**

Texto

Descrição gerada automaticamente

Imagem 13

**Não atualizável**

Uma imagem contendo Texto

Descrição gerada automaticamente

Imagem 14

# ESTUDO DO CASO

## Criação de um banco de dados fictício (exemplo: loja de e-commerce ou sistema de RH)

Interface gráfica do usuário, Texto, Aplicativo, Email

Descrição gerada automaticamente

Imagem 15

**Funções básicas**

# Diferença de Tabelas e Views no SQL - Entenda de Uma Vez!

Imagem 16

# BIBLIOGRAFIA

**LABORATORIO DE BANCO DE DADOS**

<https://sae.unb.br/cae/conteudo/unbfga/lbd/banco2_visoes.html#:~:text=Vis%C3%A3o%20em%20SQL&text=Em%20banco%20de%20dados%20relacionais,se%20fossem%20uma%20tabela%20real>

**KINGHOST**

<https://king.host/wiki/artigo/views-mysql/#:~:text=Uma%20view%20%C3%A9%20um%20objeto,do%20esquema%20f%C3%ADsico%20da%20base>.

**GPO**

<https://www.profissionaloracle.com.br/2023/07/29/guia-das-views-no-banco-de-dados-oracle-tipos-funcionalidades-e-exemplos-praticos/>

**RIC-CPS**

<https://ric.cps.sp.gov.br/bitstream/123456789/9920/1/bancodedados_2020_1_anabeatrizsantos_osbeneficiosdautilizacaodeviews.pdf>

**STACK OVERFLOW**

<https://pt.stackoverflow.com/questions/197842/view-n%C3%A3o-atualiz%C3%A1vel-em-sql-server>

**IBM**

<https://www.ibm.com/docs/pt-br/db2/11.1?topic=views-updatable>

# LISTA DE FIGURAS

Figura 1 – NORMAS ABNT, 2024. Disponível em:

<https://sae.unb.br/cae/conteudo/unbfga/lbd/imagens/view_logica.png>

Figura 2 – NORMAS ABNT, 2024. Disponível em:

<https://www.hashtagtreinamentos.com/wp-content/uploads/2022/09/Diferenca-de-Tabelas-e-Views-no-SQL-1-1024x489.png>

Figura 3, 4, 5 – NORMAS ABNT, 2024. Disponível em:

<https://www.profissionaloracle.com.br/2023/07/29/guia-das-views-no-banco-de-dados-oracle-tipos-funcionalidades-e-exemplos-praticos/>

Figura 6 – NORMAS ABNT, 2024. Disponível em:

<https://www.hashtagtreinamentos.com/wp-content/uploads/2022/09/Diferenca-de-Tabelas-e-Views-no-SQL-2-1024x485.png>