



SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL

SENAI “GASPAR RICARDO JUNIOR”

Curso

**TÉCNICO EM DESENVOLVIMENTO
DE SISTEMAS**

SQL Views

João Vittor Amorim da Silva

Sorocaba
Novembro – 2024



SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL

SENAI “GASPAR RICARDO JUNIOR”

João Vittor Amorim da Silva

SQL Views

Nesse artigo, será explicado
sobre o SQL Views e sua
Função em Bancos de Dados
Prof. Emerson Magalhães

Sorocaba
Novembro – 2024

SUMÁRIO

INTRODUÇÃO.....	4
1. SQL VIEWS E BANCOS DE DADOS	4
1.1. Definição de SQL Views e sua Função em Bancos de Dados.....	4
1.2. Importância das Views em Sistemas de Banco de Dados Relacionais	4
2. FUNDAMENTOS TEÓRICOS DAS SQL VIEWS	4
2.1. Definição e Funcionamento	4
2.2. Diferença entre Views e Tabelas	4
2.3. Tipos de Views	5
2.3.1. Views Simples	5
2.3.2. Views Complexas	5
2.3.3. Views Materializadas	5
3. Vantagens e Desvantagens de Usar Views.....	5
3.1. Vantagens.....	5
3.2. Desvantagens	5
4. PROCESSO DE CRIAÇÃO DE VIEWS NO SQL	6
4.1. Instrução CREATE VIEW: Sintaxe e Parâmetros.....	6
4.2. Exemplos de Views Simples e Complexas	6
4.2.1. View de Filtragem	6
4.2.2. View de Junção.....	6
4.2.3. Exemplo de View Complexa	6
4.3. Views Atualizáveis e Não Atualizáveis	6
4.3.1. Explicação e Condições para Atualização	6
4.3.2. Exemplo Prático	7
4.4. Exemplo de View para Relatório de Vendas	7
CONCLUSÃO.....	8
BIBLIOGRAFIA	Erro! Indicador não definido.

INTRODUÇÃO

1. SQL VIEWS E BANCOS DE DADOS

1.1. Definição de SQL Views e sua Função em Bancos de Dados

SQL Views são consultas salvas que geram tabelas virtuais baseadas em uma ou mais tabelas reais. Elas permitem que usuários visualizem dados específicos sem acessar diretamente as tabelas subjacentes, promovendo segurança e organização.

1.2. Importância das Views em Sistemas de Banco de Dados Relacionais

Views simplificam consultas complexas, restringem o acesso a certos dados, e mantêm a consistência em consultas repetitivas, sendo fundamentais para grandes sistemas de dados.

2. FUNDAMENTOS TEÓRICOS DAS SQL VIEWS

2.1. Definição e Funcionamento

Views são consultas pré-definidas que geram uma "tabela virtual", acessível através de uma instrução SELECT, sem armazenar dados diretamente.

2.2. Diferença entre Views e Tabelas

Enquanto tabelas armazenam dados fisicamente, views apenas refletem uma consulta, economizando espaço e facilitando o acesso a informações específicas.

2.3. Tipos de Views

2.3.1. Views Simples

Baseadas em uma única tabela, sem agregações ou junções.

2.3.2. Views Complexas

Integram dados de múltiplas tabelas, podendo incluir funções de agregação e junções.

2.3.3. Views Materializadas

Armazenam o resultado da consulta fisicamente e são atualizadas periodicamente, melhorando o desempenho em consultas pesadas.

3. Vantagens e Desvantagens de Usar Views

3.1. Vantagens

- Simplificação de Consultas: evita repetição de queries complexas.
- Segurança: permite controlar o acesso a dados confidenciais, limitando visões de dados por usuário ou função.
- Manutenção e Reutilização: facilita a manutenção e reutilização de queries comuns.

3.2. Desvantagens

- Impacto no Desempenho: views complexas podem afetar a velocidade de resposta.
- Limitações de Atualização: não é possível modificar dados diretamente em algumas views complexas.
- Manutenção de Views Materializadas: exige atualização periódica, o que pode aumentar o custo de manutenção.

4. PROCESSO DE CRIAÇÃO DE VIEWS NO SQL

4.1. Instrução CREATE VIEW: Sintaxe e Parâmetros

```
CREATE VIEW view_name AS  
SELECT columns FROM table_name  
WHERE condition;
```

4.2. Exemplos de Views Simples e Complexas

4.2.1. View de Filtragem

```
CREATE VIEW CustomerInfo AS  
SELECT Name, Email FROM Customers WHERE Active = 1;
```

4.2.2. View de Junção

```
CREATE VIEW OrderDetails AS  
SELECT Orders.OrderID, Customers.Name  
FROM Orders  
JOIN Customers ON Orders.CustomerID = Customers.CustomerID;
```

4.2.3. Exemplo de View Complexa

Combina várias tabelas e inclui agregações e filtros específicos.

4.3. Views Atualizáveis e Não Atualizáveis

4.3.1. Explicação e Condições para Atualização

Uma view é atualizável se cumprir certos requisitos, como ser baseada em uma única tabela sem funções de agregação ou DISTINCT.

4.3.2. Exemplo Prático

```
CREATE VIEW ActiveEmployees AS  
SELECT * FROM Employees WHERE Status = 'Active';
```

4.4. Exemplo de View para Relatório de Vendas

```
CREATE VIEW SalesReport AS  
SELECT ProductName, SUM(Quantity) AS TotalSold  
FROM Orders JOIN OrderDetails ON Orders.OrderID =  
OrderDetails.OrderID  
GROUP BY ProductName;
```

CONCLUSÃO

SQL Views são ferramentas poderosas para organizar, simplificar e proteger dados em sistemas SQL, oferecendo flexibilidade e eficiência. Views são essenciais em grandes sistemas por permitirem personalização de acesso e simplificação de consultas. É recomendável avaliar o desempenho e a necessidade de manutenção ao decidir usar views materializadas. Use views para consultas repetitivas e complexas; considere views materializadas em cenários de leitura intensiva; e evite views excessivamente complexas para operações que exijam atualizações diretas.