

Visão (View)



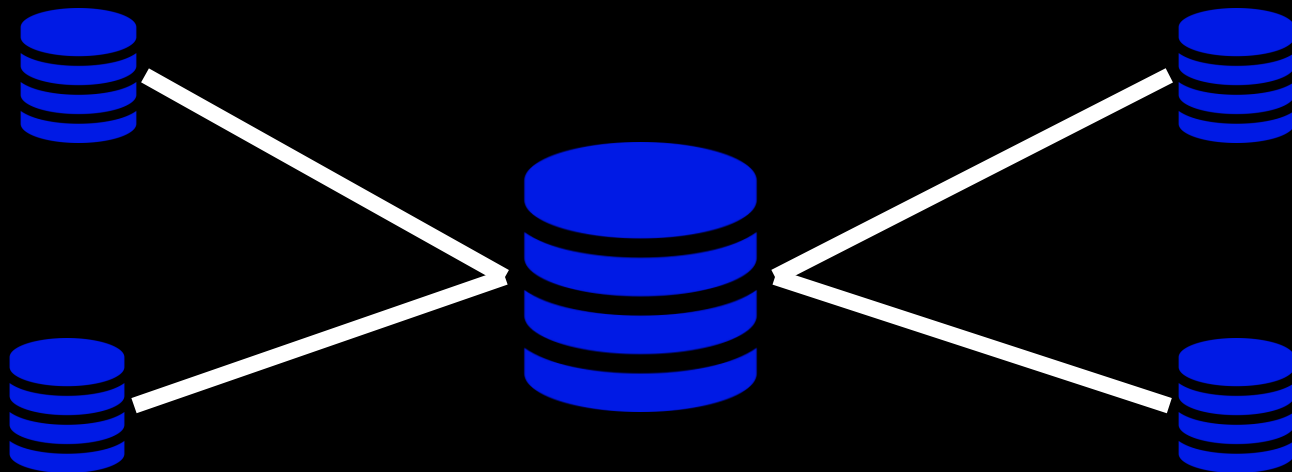
CONCEITO - VISÃO

- **O QUE É UMA VISÃO (VIEW)?**
- É uma tabela virtual, na qual os dados não estão fisicamente armazenados
- **CARACTERÍSTICAS DE UMA VISÃO**
- NÃO EXISTE fisicamente
- Considerada uma TABELA VIRTUAL
- **TABELA BASE x TABELA VIRTUAL**
- Tabela Base: tuplas (linhas) estão fisicamente armazenadas no BD
- Tabela Virtual (Views): somente a estrutura (esquema) é armazenado no banco e não as tuplas



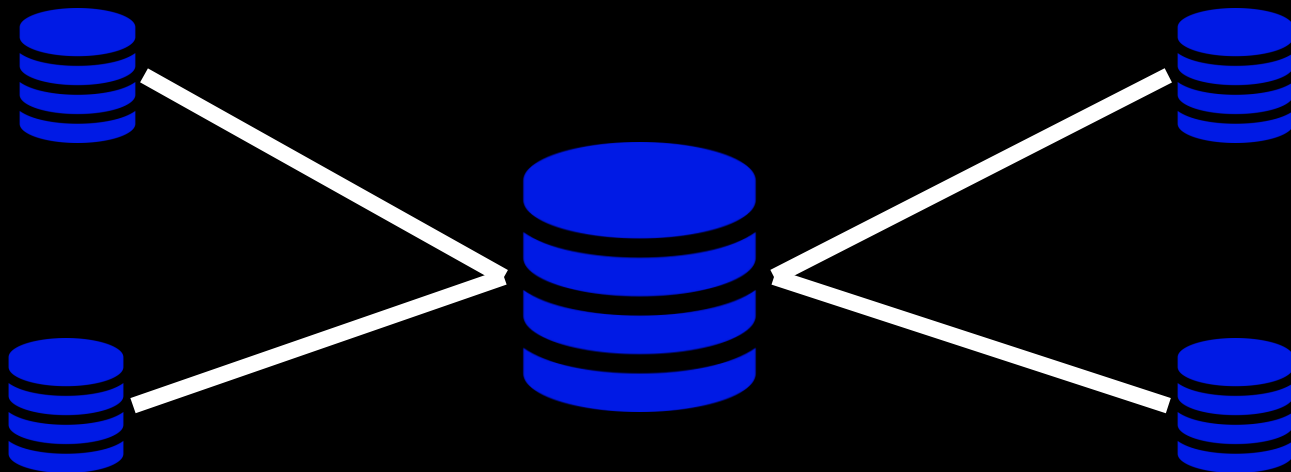
CONCEITO - VISÃO

- **PARA QUE USAR VISÕES (VIEWS)?**
- Deve-se usar visões quando for frequente consultas com junção de várias relações (tabelas), bastando definir estas junções como uma visão e posteriormente consulta-la, na cláusula FROM.



CONCEITO - VISÃO

- **PARA QUE USAR VISÕES (VIEWS)?**
- Deste modo, o objetivo é simplificar a especificação de consultas frequentes no Banco de Dados.



ESPECIFICAÇÃO DE VIEWS

- Comando CREATE VIEW

- Sintaxe:

```
CREATE VIEW <nome_visão>  
AS <consulta>
```

- **OBS.:** CONSULTA se refere a uma consulta SQL no formato tradicional: SELECT/ FROM/ WHERE (por exemplo).

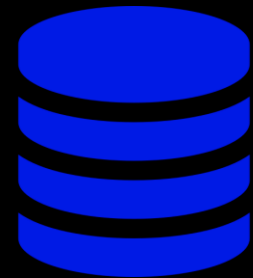
ESPECIFICAÇÃO DE VIEWS

- Após ser criada, uma visão pode ser consultada normalmente, na cláusula FROM, como se fosse uma relação (tabela) base.
- Outras visões podem ser criadas a partir das visões já definidas (visões sobre visões).



ESPECIFICAÇÃO DE VIEWs

- Comando CREATE VIEW
- RESULTADO
- Quais tuplas farão parte desta visão?
- As tuplas serão as mesmas presentes na(s) tabela(s) base resultantes da consulta usada na definição da visão.



Considere as seguintes tabelas

- CIA_AEREA

<u>CODIGO</u>	NOME	SEDE	PRESIDENTE
1	VARIG	VITORIA	RAUL SIDNEI
2	AIR FRANCE	PARIS	ETOILE ARNS
3	TAM	SÃO PAULO	CIDA FARIA
4	GOL	RIO DE JANEIRO	BIA FALCAO

- VOO

<u>NUM_VOO</u>	TARIFA	PARTIDA	DESTINO	COD_CIA*
AF234	726.20	BRASÍLIA	PARIS	2
VG893	254.30	BH	RS	1
AF875	541.00	SÃO PAULO	PARIS	2
VG487	189.40	BRASILIA	CURITIBA	1
GO142	349.00	RJ	RECIFE	4
TA961	99.00	BH	BRASILIA	3

Uso de VIEWS

- Exemplo 1:
- Recuperar o nome da companhia aérea e os voos que cada uma oferece.

```
SELECT CA.NOME, V.NUM_VOO  
FROM VOO V  
INNER JOIN CIA_AEREA CA  
ON CA.CODIGO = V.COD_CIA;
```

NOME	NUM_VOO
VARIG	VG487
VARIG	VG893
AIR FRANCE	AF234
AIR FRANCE	AF875
TAM	TA961
GOL	GO142

Uso de VIEWS

- SE o RESULTADO da consulta anterior FOSSE DEFINIDO COMO UMA VISÃO **VOO_CIA**, por exemplo, a consulta seria bem mais simples, sem junções:
- `SELECT * FROM VOO_CIA; --visão`
- RESULTADO: Simplificação das consultas que são frequentes na aplicação.
- VANTAGEM: caso a consulta seja recorrente na aplicação, é interessante utilizar visões (views).

Uso de VIEWS

--Criando a view

```
CREATE VIEW VOO_CIA AS  
SELECT NOME, NUM_VOO  
FROM VOO V  
INNER JOIN CIA_AEREA CA  
ON CA.CODIGO = V.COD_CIA;
```

--Usando a view

```
SELECT * FROM VOO_CIA;
```

Consultas



Auxilia em processos de **autorização/segurança**.

Exemplo: a empresa poderia não querer que o público tivesse acesso às tarifas dos voos (devido a uma promoção), na qual vários voos teriam o mesmo preço anunciado separadamente.

Solução: para ‘esconder’ a coluna TARIFAS, eu poderia criar uma visão da tabela VOO, que contém todos os dados, exceto TARIFAS.

Assim, o acesso não será mais à tabela base VOO e sim, à visão criada a partir da tabela VOO.

Considere a seguinte tabela

VOO

<u>NUM_VOO</u>	TARIFA	PARTIDA	DESTINO	COD_CIA*
AF234	726.20	BRASÍLIA	PARIS	2
VG893	254.30	BH	RS	1
AF875	541.00	SÃO PAULO	PARIS	2
VG487	189.40	BRASILIA	CURITIBA	1
GO142	349.00	RJ	RECIFE	4
TA961	99.00	BH	BRASILIA	3

Uso de VIEWS

EXEMPLO

```
CREATE VIEW VOO_SEM_TARIFA AS  
SELECT NUM_VOO, PARTIDA, DESTINO  
FROM VOO
```

--Consultar a visão VOO_SEM_TARIFA

```
SELECT *  
FROM VOO_SEM_TARIFA
```

num_voo	partida	destino
AF234	BRASILIA	PARIS
AF875	SAO PAULO	PARIS
GO142	RJ	RECIFE
TA961	BH	BRASILIA
VG487	BRASILIA	CURITIBA
VG893	BH	RS

Uso de VIEWS

Renomeando COLUNAS

É possível especificar os nomes das colunas da visão.

Sintaxe:

```
CREATE VIEW <nome_visão> (nome_coluna)  
AS <consulta>
```

Considere as seguintes tabelas

- CIA_AEREA

<u>CODIGO</u>	NOME	SEDE	PRESIDENTE
1	VARIG	VITORIA	RAUL SIDNEI
2	AIR FRANCE	PARIS	ETOILE ARNS
3	TAM	SÃO PAULO	CIDA FARIA
4	GOL	RIO DE JANEIRO	BIA FALCAO

- VOO

<u>NUM_VOO</u>	TARIFA	PARTIDA	DESTINO	COD_CIA*
AF234	726.20	BRASÍLIA	PARIS	2
VG893	254.30	BH	RS	1
AF875	541.00	SÃO PAULO	PARIS	2
VG487	189.40	BRASILIA	CURITIBA	1
GO142	349.00	RJ	RECIFE	4
TA961	99.00	BH	BRASILIA	3

Considere as seguintes tabelas

- Renomeando COLUNAS
- EXEMPLO: Recuperar o nome da companhia aérea e o número de voos que cada uma oferece.

```
CREATE VIEW TOTAL_VOOS  
(CIA_AEREA, NUMERO_VOOS) AS  
SELECT NOME, COUNT(NUM_VOO)  
FROM VOO V  
JOIN CIA_AEREA CA  
ON CA.CODIGO = V.COD_CIA  
GROUP BY NOME;
```

CIA_AEREA	NUMERO_VOOS
AIR FRANCE	2
GOL	1
TAM	1
VARIG	2

Considere as seguintes tabelas

- EXEMPLO 2: Selecionar as COMPANHIAS que oferecem dois ou mais voos

--basta fazer uma consulta à visão

```
SELECT *  
FROM TOTAL_VOOS  
WHERE numero_voos >= 2;
```

CIA_AEREA	NUMERO_VOOS
AIR FRANCE	2
VARIG	2

OBS.: numero_voos é o nome da coluna
count(num_voos) renomeada!

Atualização de VIEW's

O SGBD não garante que a visão sempre atualizará.

Uma view com uma única tabela de definição é atualizável.

```
UPDATE <nome_visão> SET <valores> WHERE  
<condicoes>
```

As views definidas sobre **múltiplas tabelas** usando junções **geralmente não são atualizáveis**.

As views definidas usando **funções de agrupamento** (GROUP BY) e agregação (SUM, COUNT, MAX, MIN, AVG) **não são atualizáveis**.

Atualização de VIEWS

- CONSIDERAÇÕES:
- A visão não é montada no instante da sua **definição**, mas sim no momento em que alguma **consulta** for realizada sobre ela (pois é uma tarefa custosa, em termos de desempenho, manter todas as tuplas se elas não estão sendo usadas em nenhuma consulta).
- Em outras palavras, **apenas o esquema da visão é armazenado**. Todas as vezes que a VISÃO for CONSULTADA, o SGBD calcula quais tuplas seriam retornadas por aquela consulta e exibe na tela.

Atualização de VIEWS

- EXEMPLO: -
- A CONSULTA À visão VOO_SEM_TARIFA, retorna:

num_voo	partida	destino
AF234	BRASILIA	PARIS
AF875	SAO PAULO	PARIS
GO142	RJ	RECIFE
GO542	BRASILIA	BELÉM
TA961	BH	BRASILIA
VG487	BRASILIA	CURITIBA
VG893	BH	RS

- Na tabela base VOO também será atualizado, porém o campo TARIFA terá valor NULL.

Atualização de VIEWS

- EXEMPLO:
- A CONSULTA À VISÃO
- `SELECT * FROM VOO_SEM_TARIFA`
- ANTES RETORNAVA:

num_voo	partida	destino
AF234	BRASILIA	PARIS
AF875	SAO PAULO	PARIS
GO142	RJ	RECIFE
TA961	BH	BRASILIA
VG487	BRASILIA	CURITIBA
VG893	BH	RS

- `INSERT INTO VOO_SEM_TARIFA`
- `(num_voo, partida, destino)`
- `VALUES (GO542, 'BRASILIA', 'BELEM')`

Exclusão de VIEWS

- COMANDO DROP VIEW
- Ela ficará disponível até que seja executado o comando para deletá-la .
- É possível excluir visões quando elas não forem mais necessárias; desta forma elas não farão parte do SGBD. Sintaxe:

```
DROP VIEW <nome_visão>
```

- Exemplo:
- DROP VIEW VOO_CIA;
- DROP VIEW VOO_SEM_TARIFA;

Alterar o nome da VIEW

- Comando para renomear uma VIEW
- `ALTER VIEW nome_visão RENAME TO novo_nome`

Para mudar o código da VIEW

- Utilize o **CREATE OR REPLACE VIEW**
- **CREATE OR REPLACE VIEW teste AS**
- **SELECT ***
- **FROM VOO**
- **WHERE tarifa >= 500.00;**