

Eixo Tecnológico: GESTÃO**Curso: Superior de Tecnologia em Análise e Desenvolvimento de Sistemas****Unidade Curricular / Unidade de Estudo: BANCO DE DADOS II - Manhã****Docente: Fábio Giulian Marques** prof.fabiomarques@gmail.com**Módulo/Semestre: 3º SEMESTRE****Visões (views)**

\dp – visualizar views e outros objetos

Que são VIEWS?

São uma maneira simples de executar e exibir dados selecionados de consultas complexas em bancos. Em que elas são úteis? Elas economizam grande quantidade de digitação e esforço e apresentam somente os dados que desejamos.

Criando Uma View

CREATE VIEW recent_shipments

```
AS SELECT count(*) AS num_shipped, max(ship_date), title
FROM shipments
JOIN editions USING (isbn)
NATURAL JOIN books AS b (book_id)
GROUP BY b.title
ORDER BY num_shipped DESC;
```

Usando Uma View

SELECT * FROM recent_shipments;

SELECT * FROM recent_shipments

```
ORDER BY max DESC
LIMIT 3;
```

Destruidando Uma View

DROP VIEW nomeview;

Criar as Tabelas que servirão de Base

CREATE TABLE client (

clientid SERIAL PRIMARY KEY,

clientname VARCHAR(255)

);

CREATE TABLE clientcontact (

contactid SERIAL NOT NULL PRIMARY KEY,

clientid int CONSTRAINT client_contact_check REFERENCES client(clientid),

name VARCHAR(255),

Eixo Tecnológico: GESTÃO**Curso: Superior de Tecnologia em Análise e Desenvolvimento de Sistemas****Unidade Curricular / Unidade de Estudo: BANCO DE DADOS II - Manhã****Docente: Fábio Giulian Marques** prof.fabiomarques@gmail.com**Módulo/Semestre: 3º SEMESTRE**`phone VARCHAR(255),``fax VARCHAR(255),``emailaddress VARCHAR(255)``);``CREATE VIEW client_contact_list AS``SELECT client.clientid, clientname, name, emailaddress FROM client, clientcontact``WHERE client.clientid = clientcontact.clientid;`

Estando no psql e digitando \d podemos visualizar também as views.

O nome da visão deve ser distinto do nome de qualquer outra visão, tabela, seqüência ou índice no mesmo esquema. A visão não é materializada fisicamente. Em vez disso, a consulta é executada toda vez que a visão é referenciada em uma consulta. Fazer livre uso de visões é um aspecto chave de um bom projeto de banco de dados SQL.

As visões podem ser utilizadas em praticamente todos os lugares onde uma tabela real pode ser utilizada. Construir visões baseadas em visões não é raro. Atualmente, as visões são somente para leitura: o sistema não permite inserção, atualização ou exclusão em uma visão. É possível obter o efeito de uma visão atualizável criando regras que reescrevem as inserções, etc. na visão como ações apropriadas em outras tabelas. Para obter informações adicionais consulte o comando CREATE RULE.

CREATE VIEW vista AS SELECT 'Hello World';

é ruim por dois motivos: o nome padrão da coluna é ?column?, e o tipo de dado padrão da coluna é unknown. Se for desejado um literal cadeia de caracteres no resultado da visão deve ser utilizado algo como

CREATE VIEW vista AS SELECT text 'Hello World' AS hello;

Supondo que uma consulta seja de particular interesse para uma aplicação, mas que não se deseja digitar esta consulta toda vez que for necessária, então é possível criar uma view baseada na consulta, atribuindo um nome a esta consulta pelo qual será possível referenciá-la como se fosse uma tabela comum.

`CREATE VIEW minha_view AS`

```
SELECT cidade, temp_min, temp_max, prcp, data, localizacao
FROM clima, cidades
WHERE cidade = nome;
```

`SELECT * FROM minha_visao;`

Fazer livre uso de visões é um aspecto chave de um bom projeto de banco de dados SQL. As visões permitem encapsular, atrás de interfaces que não mudam, os detalhes da estrutura das tabelas, que podem mudar na medida em que as aplicações evoluem.

As visões podem ser utilizadas em praticamente todos os lugares onde uma tabela real pode ser utilizada. Construir visões baseadas em visões não é raro.