SQL

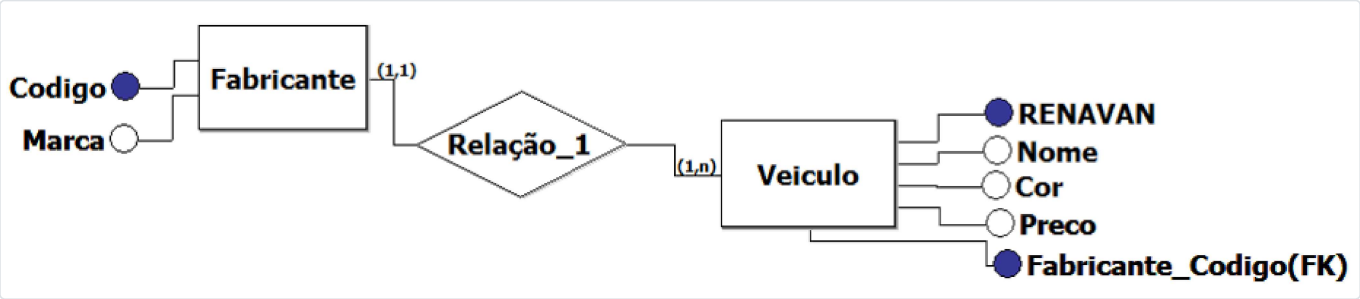
Visões e índices

Você sabia que seu material didático é interativo e multimídia? Isso significa que você pode interagir com o conteúdo de diversas formas, a qualquer hora e lugar. Na versão impressa, porém, alguns conteúdos interativos ficam desabilitados. Por essa razão, fique atento: sempre que possível, opte pela versão digital. Bons estudos!

Devido à complexidade dos bancos de dados, e a quantidade de dados inseridos nas tabelas, as consultas realizadas podem sofrer um comprometimento no tempo de processamento. Portanto, nesta webaula vamos entender as VIEWs (visões), os INDEX (índices), e as buscas textuais com o FULLTEXT, que são as técnicas presentes na linguagem de programação de banco de dados SQL.

Contextualização

Para contextualizarmos as aplicações dos conceitos, vamos tomar o exemplo de banco de dados representado no diagrama de entidade relacionamento (DER) a seguir:



Fonte: elaborada pelo autor, captura de tela do software BrModelo.

Script em SQL para a implementação do banco de dados:

```sql
CREATE DATABASE Car;
USE Car;

CREATE TABLE Fabricante (
Codigo INT(3) PRIMARY KEY AUTO_INCREMENT,
    Marca CHAR(20) NOT NULL
);

CREATE TABLE Veiculo (
RENAVAN INT(8) PRIMARY KEY,
    Nome VARCHAR(30) NOT NULL,
    Cor VARCHAR (20) NOT NULL,
    Preco DECIMAL(10,2) NOT NULL,
    fabricante_Codigo INT(3) NOT NULL,
    FOREIGN KEY (fabricante_Codigo) REFERENCES Fabricante (Codigo)
);
```

Depois que o banco de dados foi desenvolvido no sistema gerenciador de banco de dados (SGBD) MySQL, os registros foram inseridos conforme ilustrado na imagem a seguir.

```
mysql> select * from Fabricante;
+--------+-------------+
| Codigo | Marca       |
+--------+-------------+
|      1 | Volk        |
|      2 | Fait        |
|      3 | Chevroles   |
|      4 | Fordys      |
|      5 | Maudi       |
|      6 | Junday      |
+--------+-------------+
6 rows in set (0.00 sec)

mysql> select * from Veiculo;
+----------+-------------+----------+----------+-------------------+
| RENAVAN  | Nome        | Cor      | Preco    | Codigo_fabricante |
+----------+-------------+----------+----------+-------------------+
|  1234567 | Cersas      | azul     | 15000.00 |                 3 |
|  1444558 | Já          | verde    | 49000.00 |                 4 |
|  2582582 | Montanha    | lilas    | 62000.00 |                 3 |
|  2589967 | Hideas      | prata    | 44000.00 |                 2 |
|  4445566 | AAR5        | azul     | 80000.00 |                 5 |
| 10102020 | Cheveiro    | preto    | 22000.00 |                 1 |
| 11111111 | EspacialFex | amarelo  | 39000.00 |                 1 |
| 11122255 | 10S         | preto    | 33000.00 |                 3 |
| 12312312 | Cersas      | rosa     | 18000.00 |                 3 |
| 12345678 | AAR3        | prata    | 44000.00 |                 5 |
| 14714714 | Jatus       | prata    | 45000.00 |                 1 |
| 22222222 | Seniel      | preto    | 18000.00 |                 2 |
| 30303030 | Estradus    | preto    | 27000.00 |                 2 |
| 33333333 | Pins        | preto    | 40000.00 |                 3 |
| 36544477 | Linearrr    | prata    | 35000.00 |                 2 |
| 44444444 | Pins        | prata    | 38000.00 |                 3 |
| 45645645 | Hideas      | branco   | 42000.00 |                 2 |
| 55220044 | Festinnn    | branco   | 25000.00 |                 4 |
| 65465465 | AAR3        | verde    | 54000.00 |                 5 |
| 66666666 | Já          | preto    | 19000.00 |                 4 |
| 74174174 | 10S         | azul     | 23000.00 |                 3 |
| 77889966 | Montanha    | preto    | 32000.00 |                 3 |
| 78889994 | Jatus       | prata    | 55000.00 |                 1 |
| 78978998 | Golos       | dourado  | 82000.00 |                 1 |
| 85285285 | Linearrr    | amarelo  | 55000.00 |                 2 |
| 87654321 | Golos       | azul     | 32000.00 |                 1 |
| 95195195 | Golos       | preto    | 18000.00 |                 1 |
| 96396396 | Festinnn    | marrom   | 25000.00 |                 4 |
| 98798798 | AAR5        | blindado | 40000.00 |                 5 |
+----------+-------------+----------+----------+-------------------+
29 rows in set (0.00 sec)
```

Fonte: elaborada pelo autor, captura de tela do software MySQL.

VIEW

O recurso SQL para gerar visões é uma alternativa para visualizar os dados de uma ou mais tabelas de um BD. Uma visão, pode ser considerada uma “tabela virtual”, ou ainda, uma consulta pré-armazenada por meio de scripts. Geralmente a técnica de VIEW encapsula uma seleção de dados (SELECT), na qual esses dados da tabela virtual são armazenados no *cache* do SGBD.

A VIEW é uma das técnicas que podem auxiliar na diminuição na carga de processamento. Ao utilizar uma VIEW para efetuar seleção de dados, as consultas tornam-se mais rápidas, e exige menos carga de processamento. Isso ocorre, porque uma VIEW não necessita fazer o retrabalho ao executar um SELECT, pois a seleção já está pré-armazenada.

A seguir, veja as sintaxes para criação, visualização, utilização e exclusão de uma VIEW:

Criação

Sintaxe utilizada para se desenvolver uma VIEW:

```
CREATE VIEW [nome_da_VIEW] AS
  SELECT [coluna]
  FROM [tabela]
  WHERE [condições];
```

Exemplo: Partindo do cenário desenvolvido, vamos desenvolver uma VIEW que selecione a marca do fabricante, o nome, a cor e o preço do veículo, quando os valores desses veículos forem menor do que R$ 50.000,00.

Sintaxe:

```
CREATE VIEW v_selecta AS
SELECT veiculo.nome as “Veiculo”, fabricante.marca as “Marca”, veiculo.cor as Cpr, veiculo.preco as “Valor”
FROM veiculo IINER JOIN fabricante
WHERE veiculo.fabricante_Codigo = fabricante.código AND veiculo.preco <= 50000;
```

Visualização

Como as VIEWs são consideradas “tabelas virtuais”, para visualizá-las, basta exibirmos as tabelas inseridas no BD.

Sintaxe:

```
SHOW TABLES
```

Resultado:

```
mysql> show tables;
+----------------+
| Tables_in_car  |
+----------------+
| fabricante     |
| v_select1      |
| veiculo        |
+----------------+
3 rows in set (0.05 sec)
```

Fonte: elaborada pelo autor, captura de tela do software MySQL.

Utilização

Para utilizarmos uma VIEW para exibir uma consulta, deve ser utilizada a sintaxe a seguir:

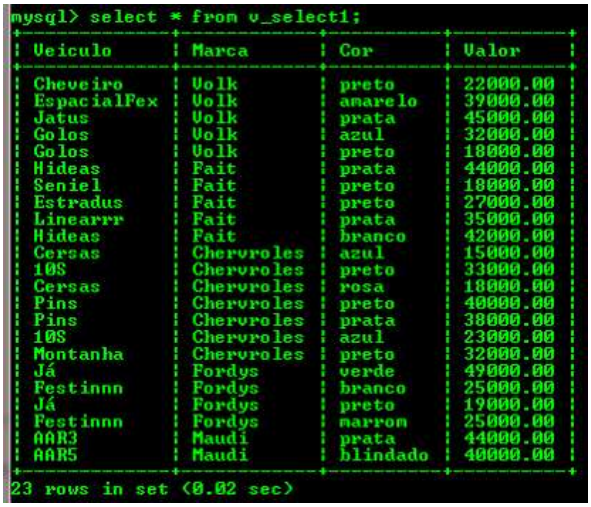
Sintaxe:

```
SELECT * FROM [nome_da_VIEW; ]
```

No exemplo desenvolvido:

```
SELECT * FROM v_select1;
```

Resultado:



```
mysql> select * from v_select1;
+-----------+-----------+----------+----------+
| Veiculo   | Marca     | Cor      | Valor    |
+-----------+-----------+----------+----------+
| Cheveiro  | Volk      | preto    | 22000.00 |
| EspacialPex | Volk    | amarelo  | 39000.00 |
| Jatus     | Volk      | prata    | 45000.00 |
| Golos     | Volk      | azul     | 32000.00 |
| Golos     | Volk      | preto    | 18000.00 |
| Hideas    | Fait      | prata    | 44000.00 |
| Seniel    | Fait      | preto    | 18000.00 |
| Estradus  | Fait      | preto    | 27000.00 |
| Linearrr  | Fait      | prata    | 35000.00 |
| Hideas    | Fait      | branco   | 42000.00 |
| Cersas    | Chevvroles | azul    | 15000.00 |
| 10S       | Chevvroles | preto   | 33000.00 |
| Cersas    | Chevvroles | rosa    | 18000.00 |
| Pins      | Chevvroles | preto   | 40000.00 |
| Pins      | Chevvroles | prata   | 38000.00 |
| 10S       | Chevvroles | azul    | 23000.00 |
| Montanha  | Chevvroles | preto   | 32000.00 |
| Já        | Fordys    | verde    | 49000.00 |
| Festinnn  | Fordys    | branco   | 25000.00 |
| Já        | Fordys    | preto    | 19000.00 |
| Festinnn  | Fordys    | marrom   | 25000.00 |
| AAR3      | Maudi     | prata    | 44000.00 |
| AAR5      | Maudi     | blindado | 40000.00 |
+-----------+-----------+----------+----------+
23 rows in set (0.02 sec)
```

Fonte: elaborada pelo autor, captura de tela do software MySQL.

Exclusão

Para excluir uma VIEW a sintaxe utilizada deve ser:

Sintaxe:

```
DROP VIEW [nome_da_VIEW];
```

INDEX

O recurso SQL para aumentar a velocidade das consultas nos bancos de dados é a utilização de índices. O recurso de índice (INDEX no MySQL) não era admitido até a versão SQL:1999. Após isso, os engenheiros buscaram um recurso para diminuir a taxa de processamento nas buscas nas tabelas, e para imposição das restrições de integridade.

A utilização dos índices é opcional para a seleção de dados, pois os índices são considerados estruturas redundantes. O SGBD pode decidir quais índices devem ser criados, porém nem sempre essa escolha automatizada, pode trazer algum benefício no processamento.

A seguir, veja as sintaxes para declaração, verificação, utilização e exclusão de um INDEX:

Declaração

- Sintaxe para declarar um índice, no desenvolvimento da tabela:

```
CREATE TABLE [nomeDaTabela] (
    Campo1 tipo(tamanho),
    Campo2 tipo(tamanho),
    INDEX(Campo1)
```

- Sintaxe para declarar um índice, em uma tabela existente no BD:

```
CREATE TABLE [nomeDoIndice] ON
    [nomeDaTabela](Campo);
```

Exemplo: Vamos criar um índice na chave primária RENAVAN (Registro Nacional de Veículos Automotores) da tabela veículo, no nosso exemplo, utilizamos a sintaxe:

```
CREATE INDEX idx_Renavam ON veiculo(RENAVAN);
```

Verificação

Sintaxe para nos certificarmos que os índices foram criados:

```
SHOW INDEX FROM [nomeDaTabela];
```

No exemplo desenvolvido:

```
SHOW INDEX FROM veiculo;
```

Resultado:

```
mysql> show index from veiculo;
+---------+------------+-------------------+--------------+-------------+
| Table   | Non_unique | Key_name          | Seq_in_index | Column_name |
+---------+------------+-------------------+--------------+-------------+
| veiculo |          0 | PRIMARY           |            1 | RENAVAN     |
| veiculo |          1 | fabricante_Codigo |            1 | fabricante_Codigo |
| veiculo |          1 | idx_Renavam       |            1 | RENAVAN     |
+---------+------------+-------------------+--------------+-------------+
3 rows in set (0.00 sec)
```

Utilização

Sintaxe:

```
SELECT [coluna] FROM [nomeDaTabela]
USE INDEX (nomeDoIndice)
WHERE [condições];
```

No exemplo desenvolvido:

```
SELECT nome AS "Veiculo", cor AS "Cor", Preço AS "Valor"
FROM veiculo
USE INDEX(idx_Renavam)
WHERE preco<= 50000;
```

Resultado:

```
+----------+---------+----------+
| Veiculo  | Cor     | Valor    |
+----------+---------+----------+
| Cersas   | azul    | 15000.00 |
| Já       | verde   | 49000.00 |
| Hideas   | prata   | 44000.00 |
| Cheveiro | preto   | 22000.00 |
| EspacialFex | amarelo | 39000.00 |
| 10S      | preto   | 33000.00 |
| Cersas   | rosa    | 18000.00 |
| AAR3     | prata   | 44000.00 |
| Jatus    | prata   | 45000.00 |
| Seniel   | preto   | 18000.00 |
| Estradus | preto   | 27000.00 |
| Pins     | preto   | 40000.00 |
| Linearrr | prata   | 35000.00 |
| Pins     | prata   | 38000.00 |
| Hideas   | branco  | 42000.00 |
| Festinnn | branco  | 25000.00 |
| Já       | preto   | 19000.00 |
| 10S      | azul    | 23000.00 |
| Montanha | preto   | 32000.00 |
| Golos    | azul    | 32000.00 |
| Golos    | preto   | 18000.00 |
| Festinnn | marrom  | 25000.00 |
| AAR5     | blindado | 40000.00 |
+----------+---------+----------+
23 rows in set (0.04 sec)
```

Fonte: elaborada pelo autor, captura de tela do software MySQL.

Exclusão

Sintaxe:

```
DROP INDEX (nomeDoIndice);
```

FULLTEXT

Outro recurso que tem uma função muito parecida com o INDEX é o FULLTEXT. Esse recurso tem a capacidade de buscar um trecho dentro de várias *strings*, assim como a função "localizar" existente nos navegadores de internet, editores de texto, etc.

Sintaxe:

```
ALTER TABLE [nome_tabela] ADD FULLTEXT (nome_da_coluna);
```

Nesse comando ao especificar uma determinada coluna como FULLTEXT, a mesma passa a ter as strings no interior de um texto monitoradas.

Para utilizar esse recurso, deve-se utilizar a sintaxe descrita a seguir:

```
SELECT [coluna] FROM nome_da_tabela
WHERE MATCH(coluna)
AGAINST('palavra_desejada');
```

Saiba Mais

Nesta webaula, pudemos compreender de que forma os recursos SQL denominados VIEW e INDEX, podem auxiliar na missão em economizar o processamento das informações quando se necessita efetuar consultas em base de dados muito extensa.

Fonte: Shutterstock.