

# Sistemas de Informação

# **Banco de Dados**

Prof. Dr. Ronaldo Castro de Oliveira

[ronaldo.co@ufu.br](mailto:ronaldo.co@ufu.br)

FACOM

---

# SQL DDL

- CREATE ...

- Instrução CREATE: utilizada para criação de estruturas no DDL, permite criar bancos, schemas, tabelas, bancos, restrições, ...

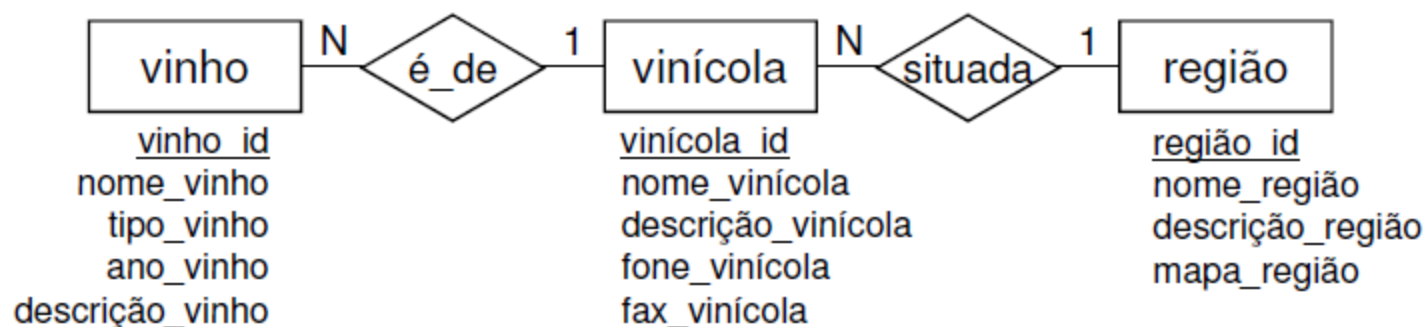
- DROP

- Instrução DROP: elimina praticamente tudo aquilo criado pelo create.

- Ambas as instruções são exclusivas do DDL

---

# Exemplo



- **região** (região\_id, nome\_região, mapa\_região, descrição\_região)
- **vinícola** (vinícola\_id, nome\_vinícola, descrição\_vinícola, fone\_vinícola, fax\_vinícola, **região\_id**)
- **vinho** (vinho\_id, nome\_vinho, tipo\_vinho, ano\_vinho, descrição\_vinho, **vinícola\_id**)

---

## Exemplo ...

```
CREATE DATABASE loja_vinhos;
```

```
CREATE TABLE região  
(  
    região_id int NOT NULL,  
    nome_região varchar(100) NOT NULL,  
    mapa_região blob,  
    descrição_região blob,  
    PRIMARY KEY (região_id),  
);
```

BLOB: *Binary Long Objects* (para armazenar áudio e vídeo)

---

---

# Exemplo

```
CREATE TABLE vinícola
(
  vinícola_id int NOT NULL,
  nome_vinícola varchar(100) NOT NULL,
  descrição_vinícola blob,
  fone_vinícola varchar(15),
  fax_vinícola varchar(15),
  região_id int DEFAULT '0' NOT NULL,
  PRIMARY KEY (vinícola_id),
  FOREIGN KEY (região_id)
    REFERENCES região (região_id)
    ON UPDATE SET DEFAULT,
    ON DELETE SET DEFAULT,
);
```

---

# Exemplo

```
CREATE TABLE vinho
```

```
(
```

```
    vinho_id int NOT NULL,
```

```
    nome_vinho varchar(50) DEFAULT '' NOT NULL,
```

```
    tipo_vinho varchar(10) DEFAULT '' NOT NULL,
```

```
    ano_vinho int DEFAULT '0' NOT NULL,
```

```
    descrição_vinho blob,
```

```
    vinícola_id int DEFAULT '0' NOT NULL,
```

```
    PRIMARY KEY (vinho_id),
```

```
    FOREIGN KEY (vinícola_id)
```

```
        REFERENCES vinícola (vinícola_id),
```

```
    ON UPDATE CASCADE
```

```
    ON DELETE CASCADE,
```

```
);
```

---

# SQL DML

- **SELECT ... FROM ... WHERE ...**
    - lista atributos de uma ou mais tabelas de acordo com alguma condição
  - **INSERT INTO ...**
    - insere dados em uma tabela
  - **DELETE FROM ... WHERE ...**
    - remove dados de tabelas já existentes
  - **UPDATE ... SET ... WHERE ...**
    - altera dados específicos de uma tabela
-

---

# Inserção

- Realizada através da especificação:
    - de uma tupla particular;
    - de uma consulta que resulta em um conjunto de tuplas a serem inseridas;
  - Valores dos atributos das tuplas inseridas:
    - devem pertencer ao domínio do atributo;
  - Atributos sem valores:
    - especificados por NULL ou valor DEFAULT;
-



---

# INSERT

```
INSERT INTO nome_tabela  
VALUES ( V1, V2, ..., VN );
```

- Ordem dos atributos deve ser mantida

---

# INSERT

```
INSERT INTO nome_tabela (A1, A2, ..., An)  
VALUES ( V1, V2, ..., VN ) ;
```

- Ordem dos atributos não precisa ser mantida

---

# INSERT

```
INSERT INTO nome_tabela  
  SELECT ...  
  FROM ...  
  WHERE ... ;
```

- Tuplas resultantes da cláusula SELECT serão inseridas na tabela nome\_tabela
-

---

## Exemplos

- INSERT INTO região  
VALUES (NULL, 'nome região', NULL,  
          'descrição');
  - INSERT INTO região (nome\_região,  
                          descrição\_região)  
VALUES 'nome região', 'descrição';
-

---

# DELETE

```
DELETE FROM nome_tabela  
WHERE predicado ;
```

## ■ Cláusula WHERE

□ é opcional:

- todas as tuplas da tabela são eliminadas
- a tabela continua a existir

---

# DELETE ...

- Remove tuplas inteiras
  - Opera apenas em uma relação
  - Tuplas de mais de uma relação a serem removidas:
    - um comando DELETE para cada relação
  - A remoção de uma tupla de uma relação pode ser propagada para tuplas em outras relações devido às restrições de integridade referencial.
-

---

## Exemplos

- **DELETE FROM vinícola**  
**WHERE vinícola\_id = 10;**
    - ❑ remove a tupla referente a vinícola\_id = 10;
    - ❑ tabela vinho (i.e., se CASCADE foi especificada na cláusula ON DELETE p/ vinícola\_id desta tabela)
  - **DELETE FROM região**
    - ❑ remove todos os dados da tabela região
-

---

# UPDATE

```
UPDATE nome_tabela  
SET coluna = <valor>  
WHERE predicado ;
```

- Cláusula WHERE
    - é opcional
  - Exemplos de <valor>
    - NULL
    - 'string'
-



---

## UPDATE ...

- Opera apenas em uma relação
  - A atualização da chave primária pode ser propagada para tuplas em outras relações devido às restrições de integridade referencial
-

---

## Exemplos

- Alterar os anos de produção de vinhos de 2005 para 2003.

UPDATE vinho

SET ano\_vinho = 2003

WHERE ano\_vinho = 2005;

- Suponha o atributo adicional **preço** na tabela vinho. Aumentar os preços dos vinhos em 10%.

UPDATE vinho SET preço = preço \* 1.10;

---

---

# Exemplos

- UPDATE vinícola

SET vinícola\_id = 10

WHERE vinícola\_id = 2;

- altera o valor de vinícola\_id = 10 para vinícola\_id = 2
    - tabela vinícola
    - tabela vinho (i.e., se a opção CASCADE foi especificada na cláusula ON UPDATE do campo vinícola\_id desta tabela)
-

**OBRIGADO A TODOS**

**DÚVIDAS**

