



Universidade Federal do Ceará
Campus Crateús

Fundamentos de Banco de Dados

Aula 08 - Primeiro Banco de Dados



Professora Vitória Regina - vitoria@crateus.ufc.br



O que estudaremos?

- Linguagem SQL.
- Tipos de dados SQL.
- Manipulação de Tabelas.



Introdução

- Linguagem padrão para os bancos de dados relacionais:
 - Presente praticamente em todos os SGBD's Relacionais;
 - Fácil migração de um SGBD para outro que usa este mesmo modelo;
 - Uma das grandes responsáveis pela grande aceitação dos SGBDR's.

Introdução

- A linguagem SQL é dividido em várias partes:
 - **DDL:** Criação, alteração e remoção de tabelas e índices.
 - **DML:** Inserção, consulta, alteração e modificação de tuplas.
 - **Manipulação de dados embutida:** Acesso ao BD através de linguagens de programação.
 - **Definição de visões:** Definição de visões a partir das tabelas criadas.

Tipos de dados SQL

- **Tipos Numéricos:**

- **INT ou INTEGER:**

- Números inteiros.

- **SMALLINT:**

- Números inteiros pequenos.

- **REAL:**

- Números reais com precisão simples.

Tipos de dados SQL

- **Tipos Numéricos:**

- **DOUBLE PRECISION:**

- Número real com precisão dupla.

- **FLOAT(n):**

- Ponto flutuante com precisão definida pelo usuário.

- **NUMERIC (p,d):**

- Ponto flutuante onde p é a precisão (total de dígitos decimais) e d é a escala (número de casas depois do ponto).

Tipos de dados SQL

- **Cadeias de caracteres:**

- **CHAR(n) ou CHARACTER(N):**

- Cadeia de caracteres de tamanho fixo;
 - Caracteres brancos são adicionados à direita caso o número de caracteres não seja preenchido.

- **VARCHAR(n):**

- Cadeia de caracteres com tamanho variável, onde n é o comprimento máximo da cadeia.

- **CHARACTER LARGE OBJECT (CLOB):**

- Usado para armazenar strings grandes, como textos e documentos.

Tipos de dados SQL

■ Cadeias de bits:

- **BIT(n):**
 - Conjunto de bits de tamanho fixo.
- **BIT VARYING(n):**
 - Conjunto de bits de tamanho variável, onde n é o tamanho máximo.
- **BINARY LARGE OBJECT (BLOB):**
 - Grandes quantidades de bits, usado para armazenar vídeos, fotos, mapas, etc.

Tipos de dados SQL

- **Tipos Booleanos:**

- **BOOLEAN:**

- Valores booleanos;
 - EM SQL, ele pode assumir os valores true, false ou unknow.

Tipos de dados SQL

- **Data e Hora:**

- **DATE:**

- Especifica uma data com o formato YYYY-MM-DD.

- **TIME:**

- Especifica uma hora com o formato HH:MM:SS.

Manipulação de Tabelas

- SQL define três comandos para a definição de dados:
 - **CREATE** (criação);
 - **ALTER** (alteração);
 - **DROP** (exclusão).

Manipulação de Tabelas

- Criando uma nova tabela:
 - Usamos o comando **CREATE TABLE**;
 - Definimos um nome para a tabela;
 - Definimos depois a lista de atributos, cada qual com seu respectivo tipo;
 - Por fim, definimos as restrições:
 - Chave primária, chave estrangeira, etc.

Linguagem de definição de dados

- Criando uma nova tabela:
 - A definição de um atributo tem a seguinte sintaxe:
 - **Nome tipo [NOT NULL] [UNIQUE] [DEFAULT valor]**
 - Onde:
 - **Nome:** corresponde ao nome do atributo;
 - **Tipo:** corresponde ao tipo de dado deste atributo (Tipo SQL);
 - **NOT NULL:** restrição que indica que o valor do atributo não pode ser nulo.

Manipulação de Tabelas

- Criando uma nova tabela:
 - A definição de um atributo tem a seguinte sintaxe:
 - **Nome tipo [NOT NULL] [UNIQUE] [DEFAULT valor]**
 - Onde:
 - **[UNIQUE]:** restrição que indica que o valor do atributo é único;
 - Usada para chaves secundárias;
 - **[DEFAULT]:** usado para definir um valor inicial para o atributo caso o valor não seja informado.

Manipulação de Tabelas

- Criando uma nova tabela:
 - Podemos especificar as seguintes restrições:
 - Chave Primária:
 - **PRIMARY KEY (atributos_chaves).**
 - Chave Estrangeira:
 - **FOREIGN KEY (atributo) REFERENCES tabela_base(atributo).**
 - Restrição de integridade:
 - **CHECK (condição).**

Manipulação de Tabelas

- Criando uma nova tabela:
 - **Exemplo:** Criar uma tabela para cada esquema de relação:
 - Fornecedor (Codigo, Nome, Cidade);
 - Peça(CodPeca, Nome, Descricao);
 - Venda(CodForn, CodP).

Manipulação de Tabelas

- Criando uma nova tabela:

- Exemplo:

```
CREATE TABLE Fornecedor(  
    Codigo INT,  
    Nome VARCHAR(20) NOT NULL,  
    Cidade VARCHAR(20) NOT NULL,  
    PRIMARY KEY (Codigo)  
)
```

Manipulação de Tabelas

- Criando uma nova tabela:

- Exemplo:

```
CREATE TABLE Peca(  
    CodFornecedor INT,  
    CodPeca INT,  
    Descricao VARCHAR(50),  
    PRIMARY KEY(CodPeca),  
    FOREIGN KEY (CodFornecedor) REFERENCES Fornecedor(Codigo)  
)
```

Manipulação de Tabelas

- Definindo chaves estrangeiras:
 - Como já vimos, usamos a restrição **"FOREIGN KEY"** é usada para a definição de uma chave estrangeira:
 - Uma restrição de integridade referencial é gerada.
 - Podemos especificar o que o SGBD deve fazer para manter a integridade referencial dos dados:
 - Numa atualização (através da restrição **ON UPDATE**);
 - Numa exclusão (através da restrição **ON DELETE**).

Manipulação de Tabelas

- Definindo chaves estrangeiras:
 - Para cada uma destas restrições, temos quatro valores possíveis:
 - **Restrict:**
 - A atualização ou remoção é rejeitada.
 - **Cascade:**
 - A atualização ou exclusão é propagada para as tuplas relacionadas.

Manipulação de Tabelas

- Definindo chaves estrangeiras:
 - Para cada uma destas restrições, temos quatro valores possíveis:
 - **Set Null:**
 - As tuplas relacionadas com a tupla a ser atualizada ou removida tem o valor do atributo chave estrangeira correspondente alterado para Null.
 - **Set Default:**
 - As tuplas relacionadas com a tupla a ser atualizada ou removida tem o valor do atributo chave estrangeira correspondente alterado para um valor default.

Manipulação de Tabelas

- Definindo chaves estrangeiras:
 - Vamos recriar uma das tabelas de nosso exemplo;

```
CREATE TABLE Peca(  
    CodFornecedor INT,  
    CodPeca INT,  
    Descricao VARCHAR(50),  
    PRIMARY KEY(CodPeca),  
    FOREIGN KEY (CodFornecedor) REFERENCES Fornecedor(Codigo) ON  
DELETE RESTRICT ON UPDATE CASCADE  
)
```

Manipulação de Tabelas

- Alterando uma Tabela:
 - Podemos alterar uma tabela usando o comando **ALTER TABLE**.
 - Ao alterar uma tabela, podemos:
 - Adicionar ou eliminar uma coluna;
 - Alterar a definição de uma coluna;
 - Adicionar ou eliminar restrições.

Manipulação de Tabelas

- Alterando uma Tabela:
 - Adicionando ou removendo colunas:
 - Podemos adicionar uma coluna através da cláusula ADD:
 - Se a tabela já foi povoada, todas as tuplas recebem um valor nulo para o novo atributo;
 - A restrição NOT NULL não pode ser aplicada neste tipo de atributo.
 - Exemplo:

ALTER TABLE Empregado **ADD** Cpf VARCHAR(15);

Manipulação de Tabelas

- Alterando uma Tabela:
 - Adicionando ou removendo colunas:
 - Podemos remover uma coluna através da cláusula DROP;
 - Podemos usar as opções CASCADE ou RESTRICT para tratar as restrições e visões que referenciam o atributo a ser excluído;
 - Exemplo:

ALTER TABLE Empregado **DROP** Salario CASCADE;

Manipulação de Tabelas

- Removendo uma Tabela:
 - Podemos remover uma tabela através do operador DROP;
 - Podemos usar as opções CASCADE ou RESTRICT para tratar as tabelas e visões que referenciam a tabela a ser excluída;
 - Exemplo:
DROP TABLE Venda CASCADE;

Manipulação de Tabelas

- Incremento automático
 - Podemos criar automaticamente o valor do campo de chave primária.
 - Podemos criar um campo de incremento automático na tabela.
 - Exemplo:

```
CREATE TABLE Fornecedor(  
    Codigo INT NOT NULL AUTO_INCREMENT,  
    Nome VARCHAR(20) NOT NULL,  
    Cidade VARCHAR(20) NOT NULL,  
    PRIMARY KEY (Codigo)  
)
```



Aula 08 - Primeiro Banco de Dados



Dúvidas?
vitoria@crateus.ufc.br