

SCC-240 Banco de Dados

Profa. Elaine Parros Machado de Sousa

Linguagem SQL - DDL

SQL - Introdução

- Linguagem declarativa \Rightarrow não procedural
- Desenvolvida e implementada pelo laboratório de pesquisa da IBM em San Jose – década de 70
- Inicialmente chamada SEQUEL (*Structured English QUery Language*)
- Criada como interface entre usuários e o primeiro SGBDR (SYSTEM R)

SQL - Introdução

- Padrão industrial que atinge grande parte do mercado de SGBDs
 - atrativos:
 - simplicidade
 - grande poder de consulta
 - padrão facilita migração

SQL - Introdução

- O padrão SQL
 - ANSI/ ISSO
 - **update 2008**
 - novos recursos de orientação a objetos
 - **SQL 2006**
 - SQL/XML
 - **SQL 2003**
 - SQL/XML
 - SQL99 (SQL3)
 - conceitos de orientação a objetos
 - padrões anteriores
 - SQL92 – SQL2
 - SQL86

SQL

- Recursos:
 - DDL – *Data Definition Language*
 - DML – *Data Manipulation Language*
 - definição de restrições de integridade
 - criação de visões (*views*)
 - especificações de segurança e autorizações
 - controle de transação
 - regras para integração com linguagens de programação
 -

SQL

- Dois conjuntos principais de comandos:
 - **DDL – *Data Definition Language*** : especificação do esquema da base de dados
 - **DML – *Data Manipulation Language*** : inserção, remoção, alteração e consultas na instância da base de dados

SQL – Alguns Operadores...

- =, <, >, <=, >=, <>
- AND, OR, NOT
- *<atributo ou expressão>* **BETWEEN**
valor1 AND valor2
 - determina se o valor de um atributo ou de uma expressão se encontra dentro de um intervalo especificado de valores
- *<atributo ou expressão>* **IS NULL**
 - determina se o valor de um atributo ou de uma expressão é nulo

SQL – Alguns Operadores...

■ LIKE

- compara partes de uma sequência de caracteres
- *atributo* **LIKE** `'%string%'`
 - **%** compara qualquer *substring* (número arbitrário de caracteres)
- *atributo* **LIKE** `'_string_ _'`
 - **_** compara qualquer caractere
- comparação *case-sensitive*

SQL – Alguns Operadores...

- **<atributo ou expressão> IN**
<conjunto valores>
 - verifica se o valor de um atributo ou de uma expressão é igual a algum dos vários valores em uma lista especificada
- **<atributo ou expressão> NOT IN**
<conjunto valores>
- . . .



DDL

DDL - Introdução

- **Alguns comandos da DDL:**

- **CREATE TABLE**
- **ALTER TABLE**
- **DROP TABLE**
- **CREATE DOMAIN**
- **ALTER DOMAIN**
- **DROP DOMAIN**
- **CREATE VIEW**
- **ALTER VIEW**
- **DROP VIEW**
- **CREATE INDEX**
- **ALTER INDEX**
- **DROP INDEX**
- **...**

Comandos DDL

- **CREATE DOMAIN** – cria um nome de domínio para especificar um tipo de dado (álías)

CREATE DOMAIN *nome* **AS** *tipo*
[<restricções de coluna>]

- facilita alterações de tipos de dados
- simplifica legibilidade

Comandos DDL

- **CREATE TABLE** - criar uma tabela, definir colunas e restrições

```
CREATE TABLE tabela (  
    atrib1 tipo [<restrições da coluna 1>] ,  
    atrib2 tipo [<restrições da coluna 2>] ,  
    . . . .  
    atribn tipo [<restrições da coluna n>] ,  
  
    [<restrições da tabela>]  
) ;
```

CREATE TABLE

- Restrições de colunas
 - NOT NULL
 - DEFAULT *valor*
 - CHECK (*condição*)

```
CREATE TABLE tabela (  
    atrib1 tipo [(tamanho)] [NOT NULL | DEFAULT valor]  
        [CHECK (condição)],  
    atrib2 tipo [(tamanho)] [NOT NULL | DEFAULT valor]  
        [CHECK (condição)],  
    ...  
);
```

CREATE TABLE

- Restrições de tabela

- CHECK (*condição*)

- PRIMARY KEY (*<atributos chave primária>*)

- UNIQUE (*<atributos chave candidata>*)

- FOREIGN KEY (*<atributos chave estrangeira>*

REFERENCES *tabelaRef* [(*<chave primária>*)]

[*<ações>*]

<instrução> *<efeito>*

ON DELETE | ON UPDATE

CASCADE | SET NULL | SET DEFAULT

SQL – Alguns tipos de dado

- **INTEGER | SMALLINT | NUMBER**
- **DECIMAL [(precision, scale)]**
 - *precision* - número total de dígitos
 - *scale* - número de dígitos depois do ponto
- **DOUBLE PRECISION | FLOAT | REAL**
- **CHAR(n)** - tamanho fixo - n caracteres
- **VARCHAR(n)** - tamanho variável
 - máximo de n caracteres
- **BLOB** – *Binary Large Object*
- **DATE | TIME | TIMESTAMP**
- ...

- CREATE TABLE

```
CREATE TABLE tabela (  
    atrib1 tipo [(tamanho)] [NOT NULL | DEFAULT valor]  
        [CHECK (condição)],  
    atrib2 tipo [(tamanho)] [NOT NULL | DEFAULT valor]  
        [CHECK (condição)],  
    ...  
    [CONSTRAINT nome da restrição  
        PRIMARY KEY (<atributos chave primária>),  
    [CONSTRAINT nome da restrição  
        UNIQUE (< atributos chave candidata>),  
    [CONSTRAINT nome da restrição  
        FOREIGN KEY (<atributos chave estrangeira>)  
        REFERENCES tabelaRef [(<chave primária>)]  
            [ON DELETE CASCADE | SET NULL | SET DEFAULT]  
            [ON UPDATE CASCADE | SET NULL | SET DEFAULT],  
    [CONSTRAINT nome da restrição  
        CHECK (condição)  
);
```

Exercício

- Criar as tabelas para o seguinte esquema

Aluno = {Nome, Nusp, Idade, DataNasc}

Professor = {Nome, NFunc, Idade, Titulação}

Disciplina = {Sigla, Nome, NCred, Professor, Livro}

Turma = {Sigla, Numero, NAlunos}

Matrícula = {Sigla, Numero, Aluno, Ano, Nota}

Comandos DDL

- **ALTER TABLE** – incluir/alterar/remover definições de colunas e restrições

ALTER TABLE *tabela* <ação>;

■ <ação>:

- **ADD** *novoAtrib tipo* [<restrições de coluna>]
- **ADD** [**CONSTRAINT nome**] <restrição de tabela>
- **DROP atributo** [**CASCADE** | **RESTRICT**]
- **DROP CONSTRAINT nome**
- **ALTER atributo DROP DEFAULT**;
- **ALTER atributo SET DEFAULT <valor>**;

... •

ALTER TABLE

- **ADD *novoAtrib* tipo [<restrições de coluna>]**
 - e os valores do novo atributo nas tuplas já existentes?
- **DROP *atributo* [CASCADE | RESTRICT]**
 - **CASCADE** – todas as visões e restrições (*constraints*) que referenciam o atributo são removidas automaticamente
 - **RESTRICT** – atributo só é removido se não houver nenhuma visão ou restrição que o referencie

Exercício

Aluno = {Nome, Nusp, Idade, DataNasc}

Professor = {Nome, NFunc, Idade, Titulação}

Disciplina = {Sigla, Nome, NCred, Professor, Livro}

Turma = {Sigla, Numero, NAlunos}

Matrícula = {Sigla, Numero, Aluno, Ano, Nota}

- Modificar as tabelas para:
 - acrescentar o atributo Cidade de Origem para a tabela Aluno
 - remover o atributo Numero da tabela Turma
 - como fica a tabela matrícula?
 - acrescentar uma restrição para que a nota de um aluno seja sempre maior ou igual a zero
 - remover uma restrição da tabela Disciplina
 - definir o valor *default* Sanca para a cidade de origem dos alunos

Comandos DDL

- **DROP TABLE** - exclui uma tabela da base de dados

DROP TABLE *tabela* [**CASCADE** | **RESTRICT**];

- **CASCADE**: todas as visões e restrições que referenciam a tabela são removidas automaticamente
- **RESTRICT**: a tabela é removida somente se não for referenciada em nenhuma restrição ou visão

Exercício

- Modificar as tabelas para:
 - remover a tabela Turma
 - como fica a tabela matrícula?

Aluno = {Nome, Nusp, Idade, DataNasc, CidadeOrigem}

Professor = {Nome, NFunc, Idade, Titulação}

Disciplina = {Sigla, Nome, NCred, Professor, Livro}

Turma = {Sigla, Numero, NAlunos}

Matrícula = {Sigla, Numero, Aluno, Ano, Nota}

```
graph TD; Aluno --> Professor; Disciplina --> Turma; Turma --> Matrícula;
```

Exercício

- Modificar as tabelas para:
 - incluir em Matrícula um atributo de frequência do aluno, cujo tipo é um domínio Frequencia.

Aluno = {Nome, Nusp, Idade, DataNasc, CidadeOrigem}

Professor = {Nome, NFunc, Idade, Titulação}

Disciplina = {Sigla, Nome, NCred, Professor, Livro}

Turma = {Sigla, Numero, NAlunos}

Matrícula = {Sigla, Numero, Aluno, Ano, Nota}

```
graph TD; Aluno["Aluno = {Nome, Nusp, Idade, DataNasc, CidadeOrigem}"]; Professor["Professor = {Nome, NFunc, Idade, Titulação}"]; Disciplina["Disciplina = {Sigla, Nome, NCred, Professor, Livro}"]; Turma["Turma = {Sigla, Numero, NAlunos}"]; Matrícula["Matrícula = {Sigla, Numero, Aluno, Ano, Nota}"]; Aluno --> Professor; Disciplina --> Turma; Turma --> Matrícula; Aluno --> Matrícula;
```


Leitura recomendada

- R. Elmasri, S. Navathe: *Fundamentals of Database Systems* – 4th Edition
 - Capítulo 8
- A. Silberschatz, H. F. Korth, s. Sudarshan: *Sistema de Banco de Dados*
 - Capítulo 4