

Banco de Dados - Relacional

Prof. Marcos Vasconcelos de Oliveira

Marcos.vasconcelos4@fatec.sp.gov.br

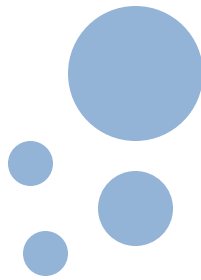


Aula 4 – Linguagem SQL – Criando Banco de Dados e Tabelas



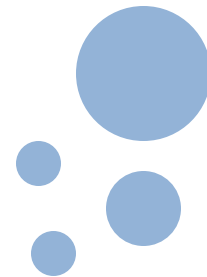
Linguagem SQL - DDL





- Create Database
- Drop Database
- Create Table
- Drop Table
- Alter Table

Create Database



O comando CREATE DATABASE cria um banco de dados

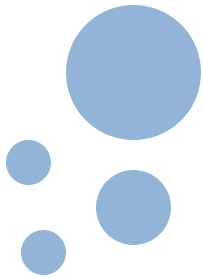
Sintaxe

```
CREATE DATABASE nome_banco_de_dados
```

nome_banco_de_dados

– Nome do banco de dados a ser criado.

Drop Database



O comando DROP DATABASE deleta um banco de dados

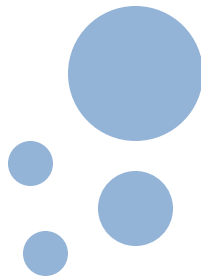
Sintaxe

```
DROP DATABASE nome_banco_de_dados
```

nome_banco_de_dados

– Nome do banco de dados a ser deletado.

Create Table



O comando CREATE TABLE cria uma tabela

Sintaxe

```
CREATE TABLE nome_tabela  
(  
  nome_coluna1 tipo_dado (tamanho),  
  nome_coluna2 tipo_dado (tamanho),  
  nome_coluna3 tipo_dado (tamanho), ...);
```

nome_tabela

– Nome da tabela a ser criada.

nome_coluna

– Nome das colunas da tabela.

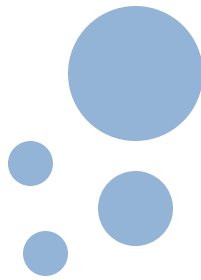
tipo_dado

– Tipo de dado para coluna

tamanho

– Especifica o tamanho máximo para o tipo de dado utilizado, especificado quando

Create Table

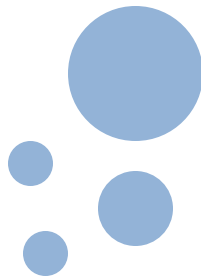


Exemplo

Cria tabela Empregado:

```
CREATE TABLE Empregado  
( CodEmpr          int ,  
  NomeEmpr         varchar(100),  
  DtNascimento     datetime  
)
```


Drop Table



O comando DROP TABLE exclui uma tabela

Sintaxe

```
DROP TABLE nome_tabela
```

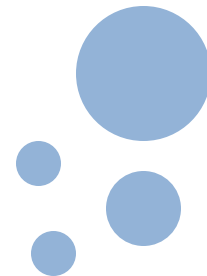
nome_tabela – Nome da tabela a ser deletada

Exemplo

Excluindo a tabela Empregado:

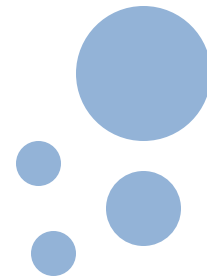
```
DROP TABLE Empregado
```

Restrições de Integridade



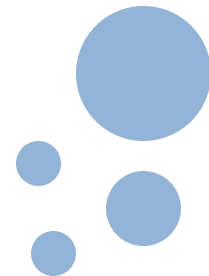
- Banco de dados Integro
 - Reflete corretamente a realidade representada pelo banco de dados e os dados são consistentes entre si
 - Um SGBD deve garantir a **integridade dos dados** através de mecanismos de **restrição de integridade**

Restrições de Integridade



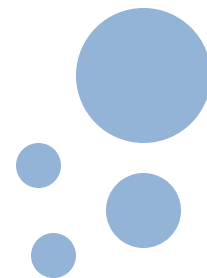
- Integridade Declarativa
 - Devem ser garantidas automaticamente por um SGBD relacional a partir da sua declaração na definição das tabelas, não sendo necessária nenhuma programação para garanti-las explicitamente
 - Chave Primária
 - Unicidade – chaves candidatas
 - Integridade Referencial – chave estrangeira
 - Integridade de Domínio
 - Tipo de dado
 - Tamanho de dado
 - Nulo
 - Intervalo de valores
 - Valores validos

Restrições de Integridade



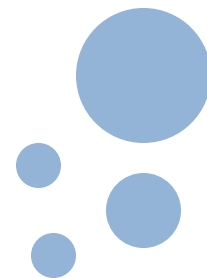
- Domínio
 - É um ***conjunto de valores*** (alfanumérico, numérico...) que um campo de uma tabela pode assumir
 - É chamado de ***domínio da coluna*** ou ***domínio do campo***

Restrições de Integridade



- Integridade de domínio
 - Especifica que o valor de um campo deve obedecer a definição de valores admitidos para a coluna – o domínio da coluna
 - Nos SGBDs relacionais é possível usar domínios pré-definidos (número inteiro, número real, alfanumérico de tamanho definido, data...)
 - É possível usar um conjunto finito de valores
 - (masculino, feminino), (sim, não), (1, 2, 5, 7, 11, 13)
 - É possível usar uma fórmula: > 0 e < 1000

Restrições de Integridade



- Valores vazios - NULL
 - Deve-se especificar se o conteúdo de uma coluna pode estar vazio (NULL)
 - Estar vazio significa que o ***campo não recebeu nenhum valor*** de seu domínio
 - As colunas que podem conter valores vazios(NULL), são as colunas ***opcionais***
 - As colunas que não podem conter valores vazios são as colunas ***obrigatórias***
- Integridade de vazio - NULL
 - Colunas que compõem a chave primária não podem ser vazios

Integridade de Domínio (Nulo) – Create Table

Sintaxe

```
CREATE TABLE nome_tabela
(  
  nome_coluna1 tipo_dado (tamanho) [NOT NULL | NULL],  
  nome_coluna2 tipo_dado (tamanho) [NOT NULL | NULL],  
  nome_coluna3 tipo_dado (tamanho) [NOT NULL | NULL]...  
)
```

nome_tabela

– Nome da tabela a ser criada.

nome_coluna

– Nome das colunas da tabela.

tipo_dado

– Tipo de dado para coluna

tamanho

– Especifica o tamanho máximo para o tipo de dado utilizado, especificado quando

quando

NOT NULL

– Define que a coluna não aceitará valores nulos

NULL

– Define que a coluna aceitará valores nulos

Integridade de Domínio (Nulo) - Create Table

Exemplo - Integridade de Domínio

Cria tabela Empregado definindo colunas como NOT NULL

```
CREATE TABLE Empregado
```

```
(  
    CodEmpr      int IDENTITY(1,1),  
    NomeEmpr     varchar(100) NOT NULL,  
    DtNascimento datetime NOT NULL,  
    CodDepto     int NOT NULL  
)
```

```
go
```


Integridade de Domínio (Nulo) –

Alter Table

```
ALTER TABLE nome_tabela  
ADD | ALTER COLUMN nome_coluna tipo_dado (tamanho) [NOT NULL | NULL]
```

nome_tabela

– Nome da tabela a ser criada.

nome_coluna

– Nome das colunas da tabela.

tipo_dado

– Tipo de dado para coluna

tamanho

– Especifica o tamanho máximo para o tipo de dado utilizado, especificado quando

quando

NOT NULL

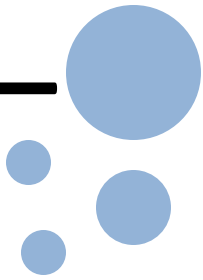
– Define que a coluna não aceitará valores nulos

NULL

– Define que a coluna aceitará valores nulos

Integridade de Domínio (Nulo) –

Alter Table



Exemplo - Nulo

Altera tabela Empregado criando e alterando colunas

```
ALTER TABLE Empregado ADD Endereco varchar(100) NOT NULL
```

```
go
```

```
ALTER TABLE Empregado ALTER COLUMN Endereco varchar(120) NULL
```

```
go
```

Integridade de Domínio (CHECK) – Create Table

Integridade de Domínio Intervalo de Valores/Valores Válidos – CHECK

Sintaxe

```
CREATE TABLE nome_tabela  
(  
  nome_coluna1 tipo_dado (tamanho) CONSTRAINT nome_check CHECK  
  [expressão],  
  nome_coluna2 tipo_dado (tamanho) CONSTRAINT nome_check CHECK  
  [expressão],  
  nome_coluna3 tipo_dado (tamanho) CONSTRAINT nome_check CHECK
```

[*expressão*]

nome_tabela

nome_coluna

tipo_dado

tamanho

quando

CONSTRAINT

nome_check

CHECK

expressão

– Nome da tabela a ser criada.

– Nome das colunas da tabela.

– Tipo de dado para coluna

– Especifica o tamanho máximo para o tipo de dado utilizado, especificado necessário.

– Define restrições de integridade

– Nome da constraint do tipo check

– Tipo de constraint

– expressão lógica que define a restrição de valores

Integridade de Domínio (CHECK) – Create Table

Exemplo

Cria tabela Empregado e define restrições de domínio:

```
CREATE TABLE Empregado
( CodEmpr          int IDENTITY(1,1) PRIMARY KEY,
  NomeEmpr         varchar(100),
  DtNascimento     datetime,
  CodDepto         int,
  Sexo             char(1) NOT NULL
                CONSTRAINT ck_EmprSexo CHECK (Sexo IN ('F','M')),
  Salario          money NOT NULL
                CONSTRAINT ck_EmprSalario CHECK (Salario > 0 And Salario < 100000)
)

go
```

Integridade de Domínio (CHECK) – Alter Table

Integridade de Domínio Intervalo de Valores/Valores Válidos – CHECK

Sintaxe

```
ALTER TABLE nome_tabela  
ADD CONSTRAINT nome_check_CHECK (expressão)
```

nome_tabela
CONSTRAINT
nome_check
CHECK
expressão

- Nome da tabela a ser criada.
- Define restrições de integridade
- Nome da constraint do tipo check
- Tipo da constraint
- expressão lógica que define a restrição de valores

Integridade de Domínio (CHECK) –

Alter Table

Exemplo

Cria tabela Empregado e define restrições de domínio, utilizando ALTER TABLE:

```
CREATE TABLE Empregado
```

```
( CodEmpr      int IDENTITY(1,1) PRIMARY KEY,
```

```
  NomeEmpr     varchar(100),
```

```
  DtNascimento datetime,
```

```
  CodDeppto    int,
```

```
  Salario      money NOT NULL,
```

```
  Sexo         char(1) NOT NULL
```

```
)
```

```
go
```

```
ALTER TABLE Empregado ADD
```

```
CONSTRAINT ck_EmprSalario CHECK (Salario > 0 And Salario < 100000)
```

```
GO
```

```
ALTER TABLE Empregado ADD
```

```
CONSTRAINT ck_EmprSexo CHECK (Sexo IN('F','M'))
```

Integridade de Domínio (CHECK) – Alter Table

Deletando Integridade de Domínio – CHECK

Sintaxe

```
ALTER TABLE nome_tabela  
DROP CONSTRAINT nome_check
```

nome_tabela

– nome da tabela que está sendo alterada

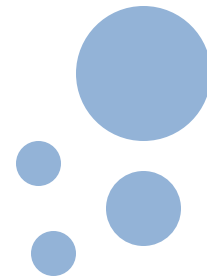
nome_chave_estrangeira

– nome da restrição de domínio que está sendo deletada

Exemplo

```
ALTER TABLE Empregado  
DROP CONSTRAINT ck_EmprSalario
```

Restrições de Integridade



- Integridade Referencial – Chave Estrangeira
 - Define que os valores dos campos que aparecem em uma chave estrangeira devem aparecer na chave primária da tabela referenciada

Integridade Referencial - Create Table

Criando Chaves Estrangeiras

Sintaxe

Opção 1 - REFERENCES

```
nome_coluna REFERENCES nome_tabela_referencia (coluna_referencia)
```

nome_coluna – nome da coluna da tabela da tabela principal

nome_tabela_referencia – nome da tabela de referencia da chave estrangeira

coluna_referencia – coluna de referencia da chave estrangeira

Opção 2 – FOREIGN KEY

```
FOREIGN KEY (nome_coluna) REFERENCES nome_tabela_referencia  
(coluna_referencia)
```

nome_coluna – nome da coluna da tabela da tabela principal

nome_tabela_referencia – nome da tabela de referencia da chave estrangeira

coluna_referencia – coluna de referencia da chave estrangeira

Integridade Referencial - Create Table

Criando Chaves Estrangeiras

Sintaxe

Opção 3 – CONSTRAINT

```
CONSTRAINT nome_chave_estrangeira FOREIGN KEY (lista_colunas) REFERENCES tabela_referencia (lista_colunas_referencia)
```

nome_chave_estrangeira

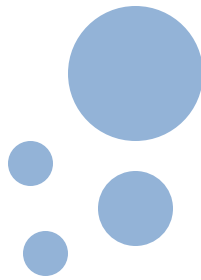
lista_colunas

tabela_referencia

lista_colunas_referencia

- nome da restrição de chave estrangeira
- lista de colunas da tabela principal
- nome da tabela de referencia das chaves estrangeiras
- lista de colunas de referencia

Create Table



Exemplo-1

Cria tabela Empregado e Departamento e referencia as duas, utilizando CONSTRAINT:

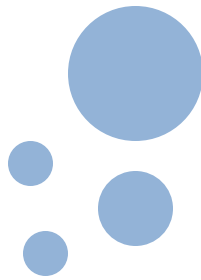
```
CREATE TABLE Departamento
```

```
(  CodDepto          int IDENTITY PRIMARY KEY,  
   NomeDepto  varchar(60)  
)  
go
```

```
CREATE TABLE Empregado
```

```
(  CodEmpr          int IDENTITY(1,1),  
   NomeEmpr        varchar(100),  
   DtNascimento    datetime,  
   CodDepto        int  
   CONSTRAINT      pk_Empregado PRIMARY KEY (CodEmpr)  
   CONSTRAINT      fk_EmprDepto FOREIGN KEY (CodDepto)  
                   REFERENCES Departamento (CodDepto)  
)
```

Create Table



Exemplo-2

Cria tabela Empregado e Departamento e referencia as duas, utilizando REFERENCES:

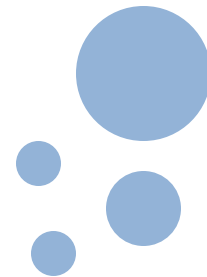
```
CREATE TABLE Departamento
```

```
( CodDepto      int IDENTITY PRIMARY KEY,  
  NomeDepto     varchar(60)  
)  
go
```

```
CREATE TABLE Empregado
```

```
( CodEmpr       int IDENTITY(1,1) PRIMARY KEY,  
  NomeEmpr      varchar(100),  
  DtNascimento  datetime,  
  CodDepto      int REFERENCES Departamento (CodDepto)  
)
```

Create Table



Exemplo-3

Cria tabela Empregado e Departamento e referencia as duas, utilizando FOREIGN KEY:

```
CREATE TABLE Departamento
( CodDepto      int IDENTITY PRIMARY KEY,
  NomeDepto     varchar(60)
)
go
```

```
CREATE TABLE Empregado
( CodEmpr       int IDENTITY(1,1) PRIMARY KEY,
  NomeEmpr      varchar(100),
  DtNascimento  datetime,
  CodDepto      int
  FOREIGN KEY (CodDepto) REFERENCES Departamento (CodDepto)
)
```

go

Integridade Referencial – Alter Table

Criando Chaves Estrangeiras

Sintaxe

```
ALTER TABLE nome_tabela  
ADD CONSTRAINT nome_chave_estrangeira FOREIGN KEY (lista_colunas)  
  REFERENCES tabela_referencia (lista_colunas_referencia)
```

nome_tabela

nome_chave_estrangeira

lista_colunas

tabela_referencia

lista_colunas_referencia

- nome da tabela que está sendo alterada
- nome da restrição de chave estrangeira
- lista de colunas da tabela principal
- nome da tabela de referencia das chaves estrangeiras
- lista de colunas de referencia

Integridade Referencial – Alter Table

Exemplo

Cria tabela Empregado e Departamento e referencia as duas, utilizando ALTER TABLE:

```
CREATE TABLE Departamento  
( CodDepcto      int IDENTITY PRIMARY KEY,  
  NomeDepcto     varchar(60)  
)  
go
```

```
CREATE TABLE Empregado  
( CodEmpr        int IDENTITY(1,1) PRIMARY KEY,  
  NomeEmpr       varchar(100),  
  DtNascimento   datetime,  
  CodDepcto      int  
)  
go
```

```
ALTER TABLE Empregado  
ADD CONSTRAINT fk_EmprDepcto FOREIGN KEY (CodDepcto)  
REFERENCES Departamento (CodDepcto)
```

Integridade Referencial – Alter Table

Deletando Chaves Estrangeiras

Sintaxe

```
ALTER TABLE nome_tabela  
DROP CONSTRAINT nome_chave_estrangeira
```

nome_tabela

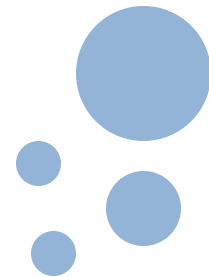
– nome da tabela que está sendo alterada

nome_chave_estrangeira

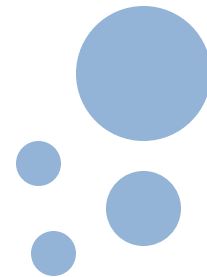
– nome da restrição de chave estrangeira que está sendo deletada

Exemplo

```
ALTER TABLE Empregado  
DROP CONSTRAINT fk_EmprDepto
```

- Atividade



Obrigado!

Prof. Marcos Vasconcelos de Oliveira