

# Structured Query Language

## Parte 1

Prof. Leandro Correia



## Bancos de Dados Relacionais

### SQL

- *Structured Query Language;*
- Linguagem de alto nível para SGBD relacionais;
- Originalmente desenvolvida pela IBM;
- Linguagem de consultas oficial padronizada pelo ANSI (*American National Standards Institute*);

## Bancos de Dados Relacionais

# SQL

- Permite armazenamento, alteração e recuperação de dados de um banco de dados, criação de bancos de dados e dos objetos nele contidos;
- Especifica **O QUE** e não **COMO**;
- Dividida em 3 categorias
  - Data Definition Language (DDL);
  - Data Manipulation Language (DML);
  - Data Control Language (DCL).

## Bancos de Dados Relacionais

### SQL

- ***Data Definition Language*** (DDL)
- Utilizada para definir bancos de dados e os objetos nele contidos;
- Exemplos de comandos DDL:
  - CREATE TABLE;
  - ALTER TABLE;
  - DROP TABLE;
  - CREATE INDEX;
  - DROP INDEX;
  - CREATE PROCEDURE.

## Bancos de Dados Relacionais

### SQL

- **Data Manipulation Language (DML);**
- Utilizada para manipular os objetos contidos no banco de dados;
- Exemplos de comandos DML:
  - SELECT;
  - INSERT;
  - UPDATE;
  - DELETE.

## Bancos de Dados Relacionais

### SQL

- **D**ata **C**ontrol **L**anguage (DCL);
- Utilizada para permitir (ou negar) o acesso aos objetos contidos no banco de dados;
- Exemplos de comandos DCL:
  - GRANT;
  - REVOKE.

# Bancos de Dados Relacionais

## SQL - DDL

```
CREATE TABLE table_name  
    ( definicao_coluna [, definicao_coluna...] [constraints] )
```

```
DROP TABLE table_name
```

- definicao\_coluna:  
col\_name tipo\_dado [NULL | NOT NULL]
- constraints:
  - PRIMARY KEY (chave primária)  
[CONSTRAINT constraint\_name] PRIMARY KEY  
[CLUSTERED | NONCLUSTERED] (col\_name [,col\_name2[,...]])
  - UNIQUE (unicidade)  
[CONSTRAINT constraint\_name] UNIQUE  
[CLUSTERED | NONCLUSTERED] (col\_name [,col\_name2[,...]])
  - FOREIGN KEY (chave estrangeira)  
[CONSTRAINT constraint\_name] FOREIGN KEY  
(col\_name [,col\_name2[,...]]) REFERENCES ref\_table  
(ref\_col\_name [,ref\_col\_name2[,...]])

## Bancos de Dados Relacionais

### SQL - DDL

```
CREATE TABLE aluno (  
    num_matricula          int          not null,  
    nome                   char(30)     not null,  
    endereco               char(40)     null,  
    telefone               char(15)     null,  
    codigo_curso           int          not null,  
  
    CONSTRAINT pk_aluno  
    PRIMARY KEY NONCLUSTERED (num_matricula),  
  
    CONSTRAINT fk_aluno_curso  
    FOREIGN KEY (codigo_curso)  
    REFERENCES curso (cod_curso)  
)
```



# Bancos de Dados Relacionais

## SQL - DDL

ALTER TABLE nome\_tabela

[ADD ( { col\_name column\_properties [column\_constraint]  
[[,] table\_constraint]} [,] next\_col\_name } ... )

| [DROP [CONSTRAINT]

constraint\_name [, constraint\_name2]...]

- EXEMPLOS:

ALTER TABLE aluno ADD data\_nascimento datetime null

ALTER TABLE aluno

CONSTRAINT fk\_aluno\_curso

FOREIGN KEY (codigo\_curso)

REFERENCES curso (cod\_curso)

ALTER TABLE aluno DROP CONSTRAINT fk\_aluno\_curso

## Bancos de Dados Relacionais

# SQL - DDL

```
CREATE [UNIQUE] [CLUSTERED | NONCLUSTERED]  
INDEX index_name ON table_name  
(column_name [, column_name] ...)
```

- EXEMPLO:

```
CREATE INDEX ix_aluno02 ON aluno (codigo_curso)
```

```
DROP INDEX table_name.index_name
```

- EXEMPLO:

```
DROP INDEX aluno.ix_aluno02
```