

# Banco de Dados

# Conceitos de SQL

- Structured Query Language, ou Linguagem de Consulta Estruturada ou SQL, é uma linguagem de pesquisa declarativa para banco de dados relacional (base de dados relacional). Muitas das características originais do SQL foram inspiradas na álgebra relacional.

# DML

- Primeiro há os elementos da DML (Data Manipulation Language - Linguagem de Manipulação de Dados). A DML é um subconjunto da linguagem usada para selecionar, inserir, atualizar e apagar dados.
  - SELECT é o comumente mais usado do DML, comanda e permite ao usuário especificar uma query como uma descrição do resultado desejado. A questão não especifica como os resultados deveriam ser localizados.
  - INSERT é usada para somar uma fila (formalmente uma tupla) a uma tabela existente.
  - UPDATE para mudar os valores de dados em uma fila de tabela existente.
  - DELETE permite remover filas existentes de uma tabela.

# DDL

- O segundo grupo é a DDL (Data Definition Language - Linguagem de Definição de Dados). Uma DDL permite ao usuário definir tabelas novas e elementos associados. A maioria dos bancos de dados de SQL comerciais tem extensões proprietárias no DDL.
- Os comandos básicos da DDL são:
  - CREATE cria um objeto (uma Tabela, por exemplo) dentro da base de dados.
  - DROP apaga um objeto do banco de dados.
- outros comandos DDL:
  - ALTER TABLE, CREATE INDEX, ALTER INDEX, DROP INDEX, CREATE VIEW e DROP VIEW

# DCL

- O terceiro grupo é o DCL (Data Control Language - Linguagem de Controle de Dados). DCL controla os aspectos de autorização de dados e licenças de usuários para controlar quem tem acesso para ver ou manipular dados dentro do banco de dados.
  - GRANT - autoriza ao usuário executar ou setar operações.
  - REVOKE - remove ou restringe a capacidade de um usuário de executar operações.
  - BEGIN WORK (ou START TRANSACTION, dependendo do dialeto SQL) pode ser usado para marcar o começo de uma transação de banco de dados que pode ser completada ou não.
  - COMMIT envia todos os dados das mudanças permanentemente.
  - ROLLBACK faz com que as mudanças nos dados existentes desde que o último COMMIT ou ROLLBACK sejam descartadas. 5

# DQL

- Embora tenha apenas um comando a DQL (Data Query Language ou Linguagem de Consulta de Dados) é a parte da SQL mais utilizada. O comando SELECT é composto de várias cláusulas e opções, possibilitando elaborar consultas das mais simples às mais elaboradas.

# Criando bases e tabelas

```
CREATE DATABASE <dbname>
```

```
CREATE SCHEMA <dbname>
```

```
CREATE TABLE <tablename> (  
    <collname> <colltype> [<collattributes>],  
    [...  
    <colnname> <colntype> [<colnattributes>]]  
)
```

```
CREATE TABLE book (  
    id INT NOT NULL PRIMARY KEY,  
    isbn VARCHAR(13),  
    title VARCHAR(255),  
    author VARCHAR(255),  
    publisher VARCHAR(255)  
)
```

# Criando índices e relacionamentos

```
CREATE INDEX <indexname>
```

```
ON <tablename> (<column1>[, ..., <columnn>])
```

```
CREATE INDEX book_isbn ON book (isbn)
```

```
CREATE TABLE book_chapter (  
    isbn VARCHAR(13) REFERENCES book (id),  
    chapter_number INT NOT NULL,  
    chapter_title VARCHAR(255)  
)
```



# Apagando objetos

**DROP TABLE** book\_chapter

**DROP SCHEMA** my\_book\_database

# Adicionando dados

```
INSERT INTO <tablename> VALUES (<field1value>[, ...,  
    <fieldnvalue>])
```

```
INSERT INTO <tablename>  
(<field1>[, ..., <fieldn>])
```

**VALUES**

```
(<field1value>[, ..., <fieldnvalue>])
```

```
INSERT INTO book (isbn, title, author)
```

```
VALUES ('0812550706', 'Ender\'s Game', 'Orson Scott  
    Card');
```

# Alterando dados

```
UPDATE book SET publisher = 'Tor Science Fiction';
```

```
UPDATE book
```

```
SET publisher = 'Tor Science Fiction', author = 'Orson S. Card'
```

```
WHERE isbn = '0812550706';
```

# Apagando dados

- **DELETE FROM** book;
- 
- **DELETE FROM** book **WHERE** isbn = '0812550706';

# Recuperando dados

```
SELECT * FROM book;
```

```
SELECT * FROM book WHERE author = 'Ray Bradbury';
```

```
SELECT * FROM book
```

```
WHERE author = 'Ray Bradbury' OR author = 'George Orwell';
```

```
SELECT * FROM book
```

```
WHERE author = 'Ray Bradbury' AND publisher LIKE '%Del Ray';
```

# Junções

```
SELECT *  
  
FROM book INNER JOIN book_chapter  
ON book.isbn = book_chapter.isbn;  
  
SELECT * FROM book  
  
INNER JOIN book_chapter  
ON book.isbn <> book_chapter.isbn;
```

# Junções

```
SELECT book.title, author.last_name  
FROM author  
LEFT JOIN book ON book.author_id = author.id;
```

```
SELECT book.title, author.last_name, publisher.name  
FROM author  
RIGHT JOIN book ON book.author_id = author.id
```

# Transações

START TRANSACTION

**DELETE FROM** book **WHERE** isbn **LIKE** '0655%'

**UPDATE** book\_chapter **set** chapter\_number = chapter\_number + 1

ROLLBACK

START TRANSACTION

**UPDATE** book **SET** id = id + 1

**DELETE FROM** book\_chapter **WHERE** isbn **LIKE** '0433%'

**COMMIT**



# Prepared Statement

- Um *prepared statement* é, essencialmente, o modelo de uma cláusula de SQL que foi pré-analisada gramaticalmente, compilada e ficou pronta para ser executada, recebendo para isso os dados apropriados.