

# SQL – MYSQL WORKBENCH/LINGUAGEM DDL

---

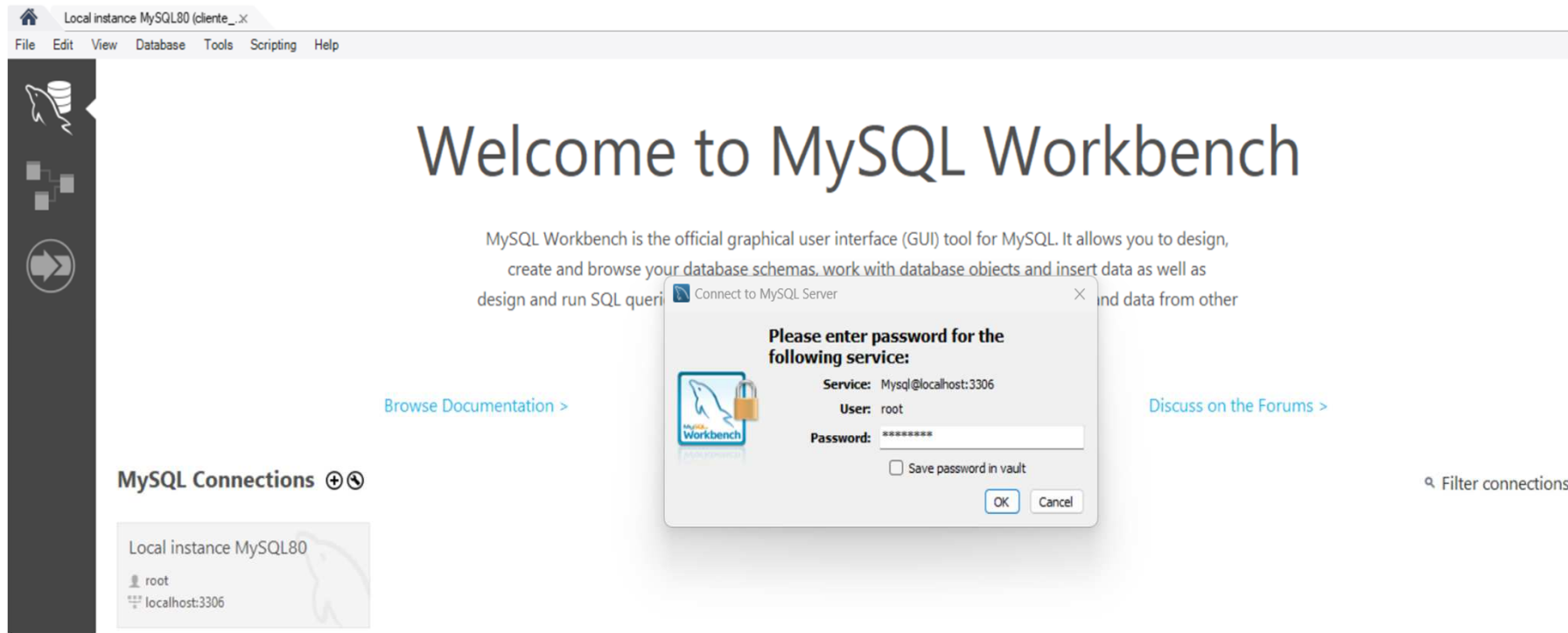
PROF. FÁBIO KRAVETZ

BANCO DE DADOS

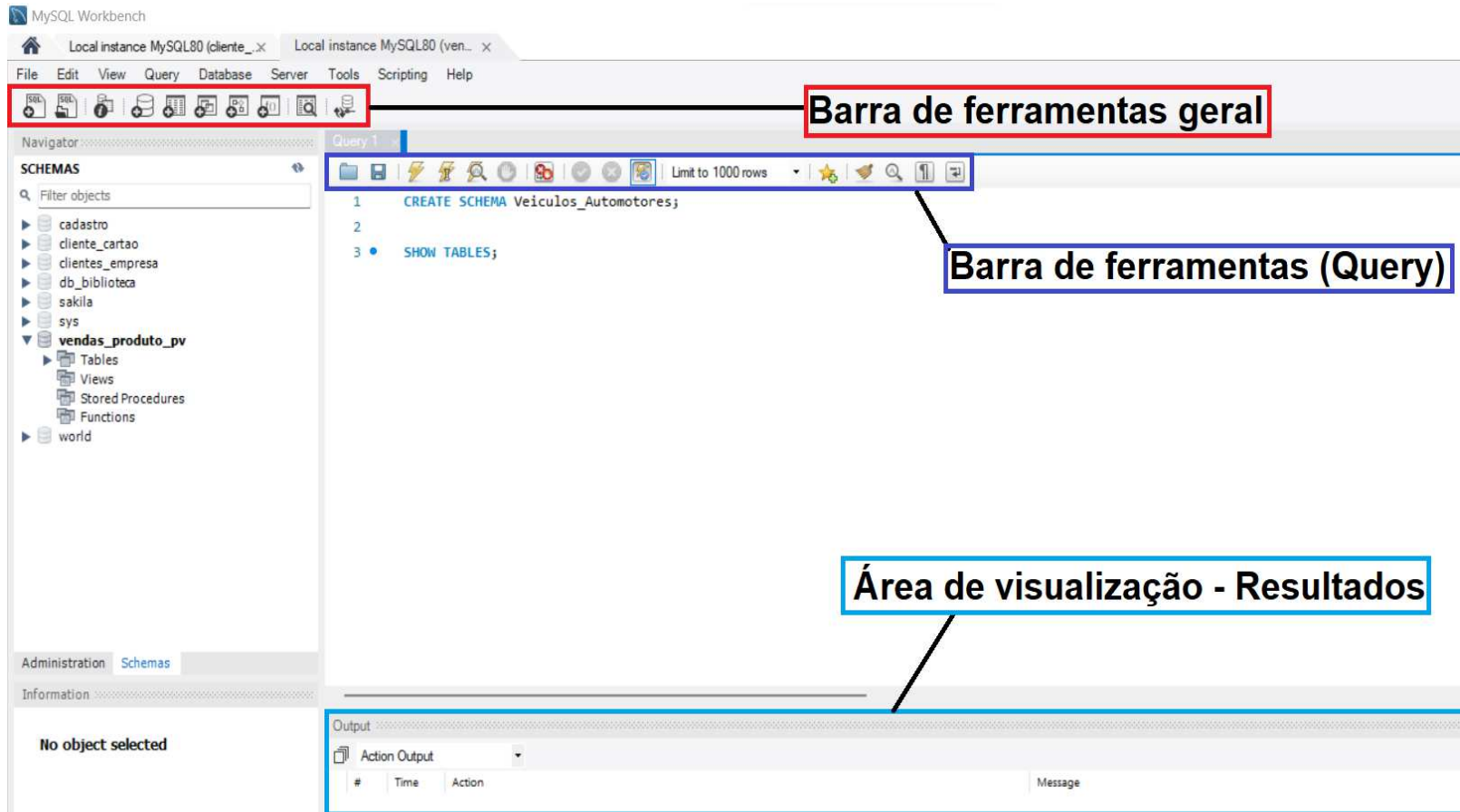
2º SEMESTRE/2023



# CONEXÃO SERVIDOR



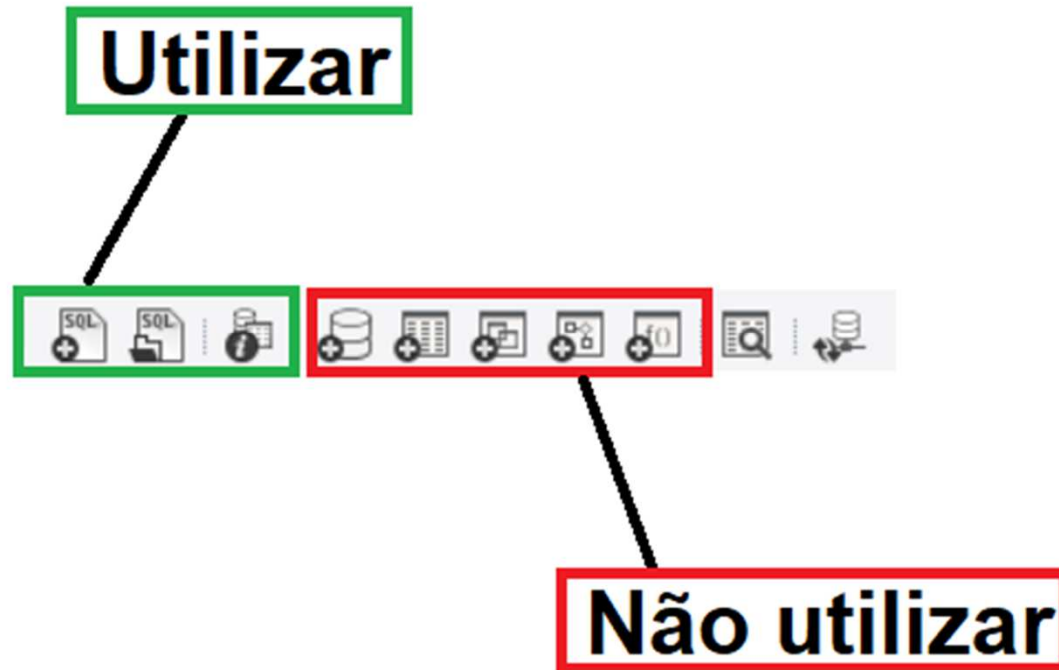
# COMPONENTES PRINCIPAIS - INTERFACE



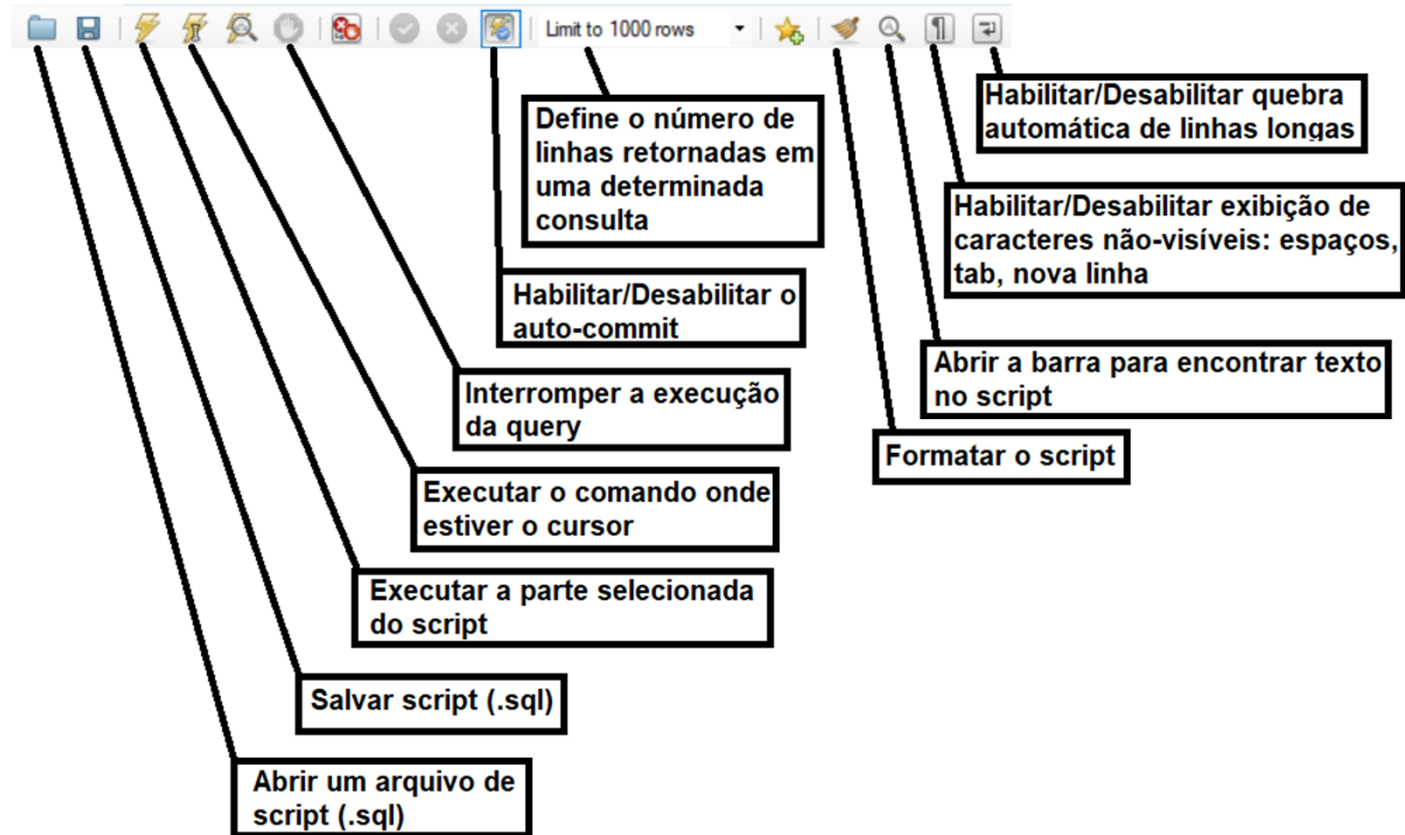
Banco de Dados - Prof. Fábio Kravetz



# BARRA DE FERRAMENTAS GERAL



# BARRA DE FERRAMENTAS (QUERY)

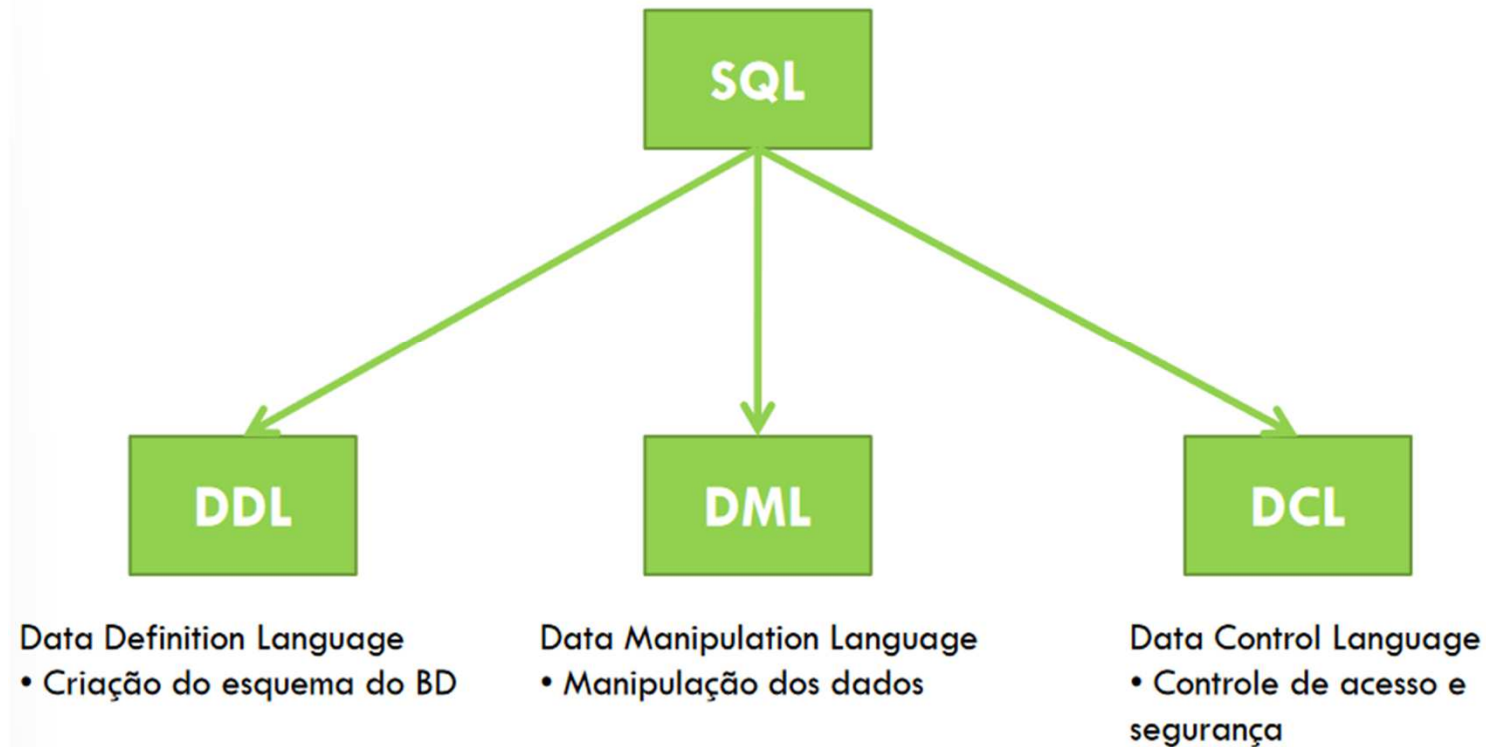


# LINGUAGEM SQL

- A SQL (*Structured Query Language*) é uma linguagem para acesso e manipulação de banco de dados relacionais;
- A SQL é uma linguagem de banco de dados abrangente, pois possui instruções para definição de dados, consultas e atualizações;
- Originalmente, SQL era chamada de SEQUEL (*Structured English QUery Language*) sendo criada pela IBM;
- Padronizada pela ANSI (*American National Standards Institute*) e pela ISO (*International Standard Organization*), tendo sua primeira versão-padrão disponibilizada em 1986 (SQL-86 ou SQL1);



# LINGUAGEM SQL



# LINGUAGEM SQL - DDL

- *Data Definition Language* é utilizada para a criação de esquemas do banco de dados;
- **As principais ações desta linguagem são:**
  - Criação de tabelas (CREATE TABLE);
  - Alteração de tabelas (ALTER TABLE);
  - Remoção de tabelas (DROP TABLE);
- Existem outras operações para manipulação dos índices:
  - CREATE INDEX; ALTER INDEX, DROP INDEX.





# LINGUAGEM SQL - DDL

- A instrução/comando CREATE TABLE é utilizada para criar uma nova tabela no banco de dados
- **CREATE TABLE** [IF NOT EXISTS] *TABLE\_NAME* (  
    *COLUMN1 DATATYPE*,  
    *COLUMN2 DATATYPE*,  
    *COLUMN3 DATATYPE*,  
    ....  
);
- **USE DATA\_BASE;**
- **CREATE TABLE** IF NOT EXISTS PESSOA (  
    ID\_PESSOA INT PRIMARY KEY AUTO\_INCREMENT,  
    NOME VARCHAR(255),  
    SOBRENOME VARCHAR (255),  
    ENDERECO VARCHAR(255),  
    CIDADE VARCHAR(255),  
    ESTADO VARCHAR(255)  
);



# LINGUAGEM SQL - DDL

- A instrução ALTER TABLE é utilizada para **adicionar**, **excluir** ou modificar colunas em uma tabela existente.
- **ALTER TABLE**  
*TABLE\_NAME*  
**ADD COLUMN\_NAME**  
*DATATYPE;*
- **ALTER TABLE** *TABLE\_NAME*  
**DROP COLUMN** *COLUMN\_NAME;*
- **ALTER TABLE** *TABLE\_NAME*  
**MODIFY COLUMN** *COLUMN\_NAME DATATYPE;*
- **ALTER TABLE** *TABLE\_NAME*  
**RENAME COLUMN** *OLD\_NAME TO NEW\_NAME;*



# LINGUAGEM SQL - DDL

- A instrução/comando ALTER TABLE é utilizada para modificar disposição dos atributos de uma determinada já existente no banco de dados
- Os principais tipos de alteração são:
  - Adição de colunas (inclusive chave primária);
  - Remoção de colunas;
  - Modificar o tipo de dado de uma coluna;
  - Renomear coluna.



# LINGUAGEM SQL - DDL

- A instrução/comando **DROP TABLE** é utilizada para **deletar/excluir** uma tabela existente de um determinado banco de dados.
- **DROP TABLE** *TABLE\_NAME*;
- A instrução/comando **TRUNCATE TABLE** é utilizada para excluir os dados dentro de uma tabela, mas não a própria tabela;
- **TRUNCATE TABLE** *TABLE\_NAME*;



# LINGUAGEM SQL - DDL

- As restrições (*constraints*) baseadas em esquema incluem restrições de domínio, restrições de chave, restrições sobre NULLs, restrições de integridade de entidade e restrições de integridade referencial (NAVATHE, 2011);
- As restrições são regras aplicadas nas colunas de uma tabela;
- São usadas para limitar os tipos de dados que são inseridos;
- Podem ser especificadas no momento de criação da tabela (CREATE) ou após a tabela ter sido criada (ALTER).



# LINGUAGEM SQL - DDL

- As principais *constraints* são:
  - NOT NULL;
  - UNIQUE;
  - PRIMARY KEY;
  - FOREIGN KEY;
  - DEFAULT.



# LINGUAGEM SQL - DDL

- **NOT NULL**

- A constraint em questão estabelece a uma determinada coluna a não aceitar valores NULL, isto é, tal restrição obriga um campo a sempre possuir um valor;

- **UNIQUE**

- A restrição UNIQUE caracteriza de forma única cada registro em uma tabela de um banco de dados;
- Uma constraint PRIMARY KEY possui automaticamente uma restrição UNIQUE definida;
- Tem se a possibilidade de ter várias constraints UNIQUE em uma mesma tabela, mas apenas uma chave primária por tabela.



# LINGUAGEM SQL - DDL

- **PRIMARY KEY**

- A restrição PRIMARY KEY (Chave Primária) identifica de forma única cada registro em uma determinada tabela do banco de dados;
- Uma coluna de chave primária não pode conter valores NULL;

- **FOREIGN KEY**

- Uma FOREIGN KEY (Chave Estrangeira) em uma tabela é um campo (coluna) que aponta para uma chave primária em outra tabela, ou seja, é usada para criar os relacionamentos entre as tabelas no banco de dados.





# LINGUAGEM SQL - DDL

- **EXEMPLO**
- `CONSTRAINT fk_ID_Autor FOREIGN KEY (ID_Autor)  
REFERENCES tbl_autores(ID_Autor)`
- Neste exemplo a chave primária está na tabela `tbl_autores` e uma chave estrangeira de nome `ID_Autor` foi criada na tabela atual (tabela livro), usando o nome `fk_ID_Autor`.



# LINGUAGEM SQL - DDL

- **DEFAULT**

- A restrição DEFAULT é usada para inserir um valor padrão especificado em uma coluna;
- Tal valor padronizado será adicionado a todos os novos registros caso nenhum outro valor seja especificado na hora de inserir dados.



# LINGUAGEM SQL - DDL

- A **chave primária** é o conceito mais básico relacionado à organização de um BD. Toda tabela, normalmente, vai possuir uma chave primária.
- Uma **chave primária** é utilizada como identificador único da tabela;
- **Chaves primárias** não podem ser nulas;
- Cada registro na tabela deve possuir uma, e somente uma, **chave primária**;
- As **chaves primárias** são incrementadas automaticamente pelo banco de dados;
  - Opção configurada no banco de dados. Caso esta opção (incremento automático) não seja definida, é necessário assegurar que não haverá valores repeditos nessa coluna.



# LINGUAGEM SQL - DDL

- `USE DATA_BASE;`
- `CREATE TABLE PESSOA (  
    ID_PESSOA INT PRIMARY KEY AUTO_INCREMENT,  
    NOME VARCHAR(255),  
    SOBRENOME VARCHAR (255),  
    ENDERECO VARCHAR(255),  
    CIDADE VARCHAR(255),  
    ESTADO VARCHAR(255)  
);`



# LINGUAGEM SQL - DDL

- A **chave estrangeira** é um conceito um pouco diferente da chave primária, pois tal identificador não diz respeito a uma determinada tabela, mas sim a um relacionamento entre tabelas;
- A **chave estrangeira** é uma referência em uma tabela a uma chave primária em outra tabela;
- **Chaves estrangeiras podem ser nulas;**
- É possível ter mais de uma (ou nenhuma) em uma tabela;
- Uma **chave estrangeira** que aceite valor *null* pode gerar o que é chamado de registro órfão, isto é, um registro sem dados para relacionamento.



# LINGUAGEM SQL - DDL

- As cláusulas FOREIGN KEY/ REFERENCES, são usualmente inseridas depois da lista de atributos, são usadas para indicar qual atributo é chaves estrangeira e qual a tabela e campo de origem.
- `USE DATA_BASE;`
- `CREATE TABLE PESSOA (  
    ID_PESSOA INT PRIMARY KEY AUTO_INCREMENT,  
    NOME VARCHAR(255),  
    SOBRENOME VARCHAR (255),  
    ENDERECO VARCHAR(255),  
    CIDADE VARCHAR(255),  
    ESTADO VARCHAR(255),  
    FOREIGN KEY (ID_CARTAO_FK)  
    REFERENCES CARTAO (ID_CARTAO)  
);`

Banco de Dados - Prof. Fábio Kravetz



# LINGUAGEM SQL - DDL

- Crie a **tabela Pessoa** e a **tabela Cartão** e seu respectivo relacionamento;
- A tabela cartão além da chave primária deve possuir os atributos número do cartão e sua respectiva validade.



# LINGUAGEM SQL - DDL

- `USE DATA_BASE;`
- `CREATE TABLE PESSOA (  
    ID_PESSOA INT PRIMARY KEY  
    AUTO_INCREMENT,  
    NOME VARCHAR(255),  
    SOBRENOME VARCHAR(255),  
    LOGRADOURO VARCHAR(255),  
    CIDADE VARCHAR(255),  
    ESTADO VARCHAR(255)  
);`
- `USE DATA_BASE;`
- `CREATE TABLE CARRO (  
    ID_CARRO INT PRIMARY KEY  
    AUTO_INCREMENT ,  
    NOME VARCHAR(255),  
    MARCA VARCHAR (255),  
    CIDADE VARCHAR(255),  
    ESTADO VARCHAR(255),  
);`





# Atividade prática

