

Linguagem SQL

Prof. Álisson R. Arantes

SQL (*Structured Query Language*)

- SQL = DDL + DML + DCL
- DDL: *Data Definition Language*
- DML: *Data Manipulation Language*
- DCL: *Data Control Language*



Shutterstock.com

DDL (*Data Definition Language*)

- Comandos DDL:

- CREATE
- ALTER
- DROP

```
CREATE TABLE Piloto (  
...  
);
```

```
ALTER TABLE Piloto ...;
```

```
DROP TABLE Piloto;
```

Pais(Numero(nn),Nome)

Piloto(Numero(nn),Nome,Pais)

Piloto[Pais] $\xrightarrow{p(r),b(m)}$ Pais[Numero]

Campeonato(Ano(nn),Campeao)

Campeonato[Campeao] $\xrightarrow{p(r),b(m)}$ Piloto[Numero]

Equipe(Numero(nn),Nome,DataFundacao,Pais)

Equipe[Pais] $\xrightarrow{p(r),b(m)}$ Pais[Numero]

Pertence(Piloto(nn),Equipe(nn),AnoEntrada,AnoSaida)

Pertence[Piloto] $\xrightarrow{p(r),b(m)}$ Piloto[Numero]

Pertence[Equipe] $\xrightarrow{p(r),b(m)}$ Equipe[Numero]

Disputa(Piloto(nn),Campeonato(nn),Pontuacao(nn))

Disputa[Piloto] $\xrightarrow{p(r),b(m)}$ Piloto[Numero]

Disputa[Campeonato] $\xrightarrow{p(r),b(m)}$ Campeonato[Ano]

CREATE DATABASE Formula1;

CREATE TABLE Pais (
Numero **INT NOT NULL**,
Nome **CHAR(20)**,
CONSTRAINT PK_Pais **PRIMARY KEY**
(Numero)
);

CREATE TABLE Piloto (
Numero **INT NOT NULL**,
Nome **CHAR(30)**,
Pais **INT**,
CONSTRAINT PK_Piloto **PRIMARY KEY**
(Numero),
CONSTRAINT FK_Piloto_Pais **FOREIGN KEY**
(Pais) **REFERENCES** Pais (Numero)
);

Autodromo(Numero(nn), Nome, Extensao, Pais)
Autodromo[Pais] $\xrightarrow{p(r), b(m)}$ Pais[Numero]

Corrida(Numero(nn), Data, Autodromo(nn), Vencedor)
Corrida[Autodromo] $\xrightarrow{p(r), p(m)}$ Autodromo[Numero]
Corrida[Vencedor] $\xrightarrow{b(r), b(m)}$ Piloto[Numero]

Participa(Piloto(nn), Corrida(nn), Posicao)
Participa[Piloto] $\xrightarrow{p(r), b(m)}$ Piloto[Numero]
Participa[Corrida] $\xrightarrow{p(r), b(m)}$ Corrida[Numero]

Banco de dados dos campeonatos de
Fórmula 1 (nível lógico - relacional)

```
CREATE TABLE Autodromo (  
  Numero INT NOT NULL,  
  Nome CHAR(30),  
  Extensao INT,  
  Pais INT,  
  CONSTRAINT PK_Autodromo PRIMARY KEY (Numero),  
  CONSTRAINT FK_Autodromo_Pais FOREIGN KEY (Pais)  
  REFERENCES Pais (Numero)  
);
```

```
CREATE TABLE Corrida (  
  Numero INT NOT NULL,  
  Data DATE,  
  Autodromo INT NOT NULL,  
  Vencedor INT,  
  CONSTRAINT PK_Corrida PRIMARY KEY (Numero),  
  CONSTRAINT FK_Corrida_Autodromo FOREIGN KEY  
  (Autodromo) REFERENCES Autodromo (Numero),  
  CONSTRAINT FK_Corrida_Vencedor FOREIGN KEY  
  (Vencedor) REFERENCES Piloto (Numero)  
);
```

Linguagem SQL (Definição dos Dados)

DEPARTAMENTO

Cod	Nome	Endereco	Orcamento
11	Pessoal	Prédio 1	10.500,00
22	Pesquisa	Prédio 2	20.000,00
33	Financeiro	Prédio 3	8.000,00
44	Manutenção	Prédio 2	5.000,00

EMPREGADO

CPF	Nome	Sexo	Salario	Depto	Supervisor
111	Antônia	F	500,00	22	222
222	Júlia	F	700,00	22	444
333	Carlos	M	900,00	33	444
444	Otávio	M	1.500,00	44	

Empregado [Depto] → Departamento[Cod]

Empregado [Supervisor] → Empregado[CPF]

Linguagem SQL (Definição dos Dados)

```
CREATE DATABASE Organizacao;
```

```
CREATE TABLE Departamento (  
Cod INT NOT NULL,  
Nome CHAR(20),  
Endereco CHAR(40),  
Orcamento DECIMAL(6,2),  
CONSTRAINT PK_Departamento PRIMARY  
KEY (Cod)  
);
```

```
CREATE TABLE Empregado (  
CPF CHAR(3) NOT NULL,  
Nome CHAR(40),  
Sexo CHAR,  
Salario DECIMAL(5,2),  
Depto INT NOT NULL,  
Supervisor CHAR(3),  
CONSTRAINT PK_Empregado PRIMARY KEY (CPF),  
CONSTRAINT FK_Empregado_Depto FOREIGN KEY  
(Depto) REFERENCES Departamento (Cod),  
CONSTRAINT FK_Empregado_Super FOREIGN KEY  
(Supervisor) REFERENCES Empregado (CPF)  
);
```

Criação do Banco de Dados Anterior em SQL

DML (*Data Manipulation Language*)

- Comandos DML:
 - INSERT
 - DELETE
 - UPDATE
 - SELECT

Linguagem SQL (Alteração dos Dados)

- Inserir o novo empregado que foi contratado (vinculado ao Departamento Financeiro e supervisionado por Carlos):
INSERT INTO Empregado **VALUES** ('888','Pedro','M',5500.00,33,'333');
- O Departamento de Manutenção está sendo extinto. Demitir todos os seus empregados:
DELETE FROM Empregado
WHERE Depto = 44;

Linguagem SQL (Alteração dos Dados)

- Aumentar em 20% o salário dos empregados do Departamento de Pesquisa:

UPDATE Empregado

SET Salario = Salario * 1.2

WHERE Depto = 22;

Linguagem SQL (Consulta aos Dados)

- Quais os nomes dos empregados do Departamento Financeiro?

SELECT E.Nome

FROM Empregado E **JOIN** Departamento D **ON** E.Depto = D.Cod

WHERE D.Nome = 'Financeiro';



Shutterstock.com

Linguagem SQL (Consulta aos Dados)

- Quais são os nomes e números de empregados de cada departamento em ordem alfabética?

```
SELECT D.Nome, COUNT(*)  
FROM Departamento D JOIN Empregado E ON E.Depto = D.Cod  
GROUP BY D.Cod, D.Nome  
ORDER BY D.Nome;
```



Shutterstock.com

Linguagem SQL (Consulta aos Dados)

- Quais os nomes dos empregados que não são supervisores de ninguém?

```
SELECT E.Nome  
FROM Empregado E  
WHERE E.CPF NOT IN (SELECT Supervisor  
                        FROM Empregado);
```



Shutterstock.com

DCL (*Data Control Language*)

- Comandos DCL:
 - GRANT
 - REVOKE



PUC Minas
Virtual