



NTT DATA BUSINESS SOLUTIONS

Customer Experience Academy

Fabiano Nascimento Leite

1

Banco de Dados Conceitos Básicos

Conceitos Básicos

■ Abordagem

- O que é banco de dados
- DDL e DML
- Instruções SQL
- Tabelas e Relações
- Exercícios e
- Exercícios

- O Sistema de Gerenciamento de Banco de Dados (**SGBD**) é um software que possui recursos capaz de manipular as informações de um banco de dados e interagir com o usuário.

- Exemplos de SGBD:
 - Oracle
 - SQL Server
 - DB2
 - PostgreSQL
 - MySQL
 - SAP HANA
 - SAP IQ
 - etc...

■ DDL – Data Definition Language

- CREATE DATABASE <nome_da_base>
- DROP DATABASE <nome_da_base>
- CREATE TABLE <nome_da_tabela>
- DROP TABLE <nome_da_tabela>
- ALTER TABLE <nome_da_tabela> ADD <nome_da_coluna> <tipo>

- DML – Data Manipulation Language

- **INSERT INTO** <nome_da_tabela> (colunas) **VALUES** (valores)
- **SELECT * FROM** <nome_da_tabela>
- **UPDATE** <nome_da_tabela> **SET** <coluna> = <valor> **WHERE** <condição>
- **DELETE FROM** <nome_da_tabela>

Conceitos Básicos

- Ideia da estrutura de um banco de dados

Tabela Pessoa

id	nome	idade
1	João	20
2	Maria	19
3	Pedro	22
4	Carlos	32
5	Ricardo	27

Tabela Estado

id	uf	nome
1	SP	São Paulo
2	PR	Paraná
3	SC	Santa Catarina
4	RJ	Rio de Janeiro

1:n

Tabela Cidade

id	nome	estado_id
1	São Paulo	1
2	Ribeirão Preto	1
3	Curitiba	2
4	Ponta Grossa	2
5	Criciúma	3
6	Angra dos Reis	4

Conceitos Básicos

- Ideia da estrutura de um banco de dados

Engenheiro

id	cpf	nome
1	111	Eng 01
2	222	Eng 02
3	333	Eng 03
4	444	Eng 04

1:n

Alocação

n:n

id	eng_id	pro_id	horas
1	1	4	2000
2	2	3	3000
3	3	2	1500
4	4	1	800

n:1

Projeto

id	nome
1	Projeto A
2	Projeto B
3	Projeto C
4	Projeto D

■ Normalização

- Normalização é o processo de modelar o banco de dados projetando a forma como as informações serão armazenadas a fim de eliminar, ou pelo menos minimizar, a redundância no banco. Tal procedimento é feito a partir da identificação de uma anomalia em uma relação, decompondo-as em relações melhor estruturadas.
- FN1: Uma relação está na primeira forma normal quando todos os atributos contém apenas um valor correspondente, singular e não existem grupos de atributos repetidos — ou seja, não admite repetições ou campos que tenham mais que um valor.
- FN2: É dito que uma tabela está na segunda forma normal se ela atende a todos os requisitos da primeira forma normal e se os registros na tabela, que não são chaves, dependam da chave primária em sua totalidade e não apenas parte dela. A segunda forma normal trabalha com essas irregularidades e previne que haja redundância no banco de dados.
- FN3: Se analisarmos uma tupla e não encontrarmos um atributo não chave dependente de outro atributo não chave, podemos dizer que a entidade em questão está na terceira forma normal - contanto que esta não vá de encontro as especificações da primeira e da segunda forma normal.

Conceitos Básicos

■ Normalização

Código	Nome	Endereço	Telefone
1001	Diego Machado	Rua Tal 321 Porto	5312345678 5398765432
1002	Fulano de Tal	Avenida Tal 71 Centro	5187654321 5143215678

Tabela 1: Tabela não está na primeira forma normal

Código	Nome	Endereço	Bairro
1001	Diego Machado	Rua Tal 321	Porto
1002	Fulano de Tal	Avenida Tal 71	Centro

Tabela 2: Tabela está na primeira forma normal

Código	Telefone
1001	5312345678
1001	5398765432
1002	5112345678
1002	5187654321

Tabela 3: Nova tabela criada para evitar campos com mais de um valor

cd_locacao	cd_filme	titulo_filme	devolucao	cd_cliente
1010	201	The Matrix	2011-10-12	743
1011	302	O Grito	2011-12-10	549
1012	201	The Matrix	2011-12-30	362

Tabela 4: Tabela não está na segunda forma normal

cd_filme	titulo_filme
201	The Matrix
302	O Grito

Tabela 5: Tabela criada para armazenar os filmes

cd_locacao	cd_filme	devolucao	cd_cliente
1010	201	2011-10-12	743
1011	302	2011-12-10	549
1012	201	2011-12-30	362

Tabela 6: Tabela na segunda forma normal

placa	modelo	qtd_kmetro	cod_fab	nome_fab
qwe1234	Modelo1	867	3004	fabricante1
asd456	Modelo2	928	3005	fabricante2

Tabela 7: Tabela não está na terceira forma normal

placa	modelo	qtd_kmetro	cod_fab
qwe1234	Modelo1	867	3004
asd456	Modelo2	928	3005

Tabela 8: Tabela na terceira forma normal

cod_fab	nome_fab
3004	fabricante1
3005	fabricante2

Tabela 9: Tabela criada para armazenar o nome do fabricante

Conceitos Básicos

- Exemplos

```
CREATE DATABASE cxacademy;
```

```
USE cxacademy;
```



■ Exemplos

```
CREATE TABLE pessoa (  
    id integer primary key auto_increment,  
    nome varchar (100),  
    cpf varchar (11)  
);  
  
CREATE TABLE telefone (  
    id integer primary key auto_increment,  
    numero varchar (100),  
    pessoa_id integer,  
    foreign key (pessoa_id) references pessoa(id)  
);
```

■ Exemplos

```
INSERT INTO pessoa (nome, cpf) VALUES ('Maria', '1111');  
INSERT INTO telefone (numero, pessoa_id) VALUES ('1111', 1);
```

```
INSERT INTO pessoa (nome, cpf) VALUES ('João', '2222');  
INSERT INTO telefone (numero, pessoa_id) VALUES ('2222', 2);
```

```
INSERT INTO pessoa (nome, cpf) VALUES ('Pedro', '3333');  
INSERT INTO telefone (numero, pessoa_id) VALUES ('3311', 3);  
INSERT INTO telefone (numero, pessoa_id) VALUES ('3322', 3);  
INSERT INTO telefone (numero, pessoa_id) VALUES ('3333', 3);
```

```
INSERT INTO pessoa (nome, cpf) VALUES ('Mary', '4444');  
INSERT INTO telefone (numero, pessoa_id) VALUES ('4444', 4);
```

■ Exemplos

```
SELECT * FROM pessoa p;  
SELECT p.nome, p.cpf, t.numero FROM pessoa p;
```

```
SELECT * FROM telefone p;  
SELECT t.numero FROM telefone t;
```

```
SELECT * FROM pessoa p JOIN telefone t ON t.pessoa_id = p.id;
```

```
SELECT p.nome AS Nome, p.cpf AS CPF, t.numero AS Telefone  
FROM pessoa p JOIN telefone t ON t.pessoa_id = p.id;
```

```
SELECT p.nome AS Nome, p.cpf AS CPF, t.numero AS Telefone  
FROM pessoa p JOIN telefone t ON t.pessoa_id = p.id  
WHERE p.cpf = '3333';
```

■ Exemplos

```
UPDATE pessoa SET nome = 'Maria Silva' WHERE cpf = '1111';
```

```
UPDATE pessoa SET nome = 'João Pedro' WHERE cpf = '2222';
```

```
UPDATE pessoa SET nome = 'Pedro Souza' WHERE cpf = '3333';
```

```
UPDATE pessoa SET nome = 'Mary Lee' WHERE cpf = '4444';
```

```
UPDATE telefone SET numero = '11-1111' WHERE id = 1;
```

```
UPDATE telefone SET numero = '9999'; - - sem WHERE
```

■ Exemplos

```
SELECT * FROM pessoa p ORDER BY nome;  
SELECT * FROM pessoa p ORDER BY nome ASC;  
SELECT * FROM pessoa p ORDER BY nome DESC;
```

```
ALTER TABLE pessoa ADD idade integer;
```

```
UPDATE pessoa SET idade = 10 WHERE id = 1;  
UPDATE pessoa SET idade = 60 WHERE id = 2;  
UPDATE pessoa SET idade = 20 WHERE id = 3;
```

```
SELECT MAX(idade) FROM pessoa;  
SELECT MIN(idade) FROM pessoa;  
SELECT AVG(idade) FROM pessoa;  
SELECT SUM(idade) FROM pessoa;  
SELECT COUNT(id) FROM pessoa;
```


■ Exemplos

```
DELETE telefone WHERE id = 1;
```

```
DELETE telefone WHERE pessoa_id = 2;
```

```
DELETE pessoa WHERE cpf = '3333'; - - relação telefone
```

```
DELETE telefone t WHERE t.pessoa_id = (SELECT p.id FROM pessoa p WHERE p.cpf = '3333');
```

```
DELETE pessoa WHERE cpf = '3333';
```

```
DELETE telefone;
```

```
DELETE pessoa;
```

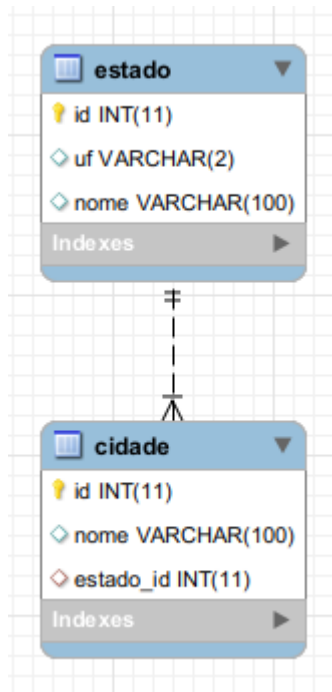
- - INSERT com novos ids

2

Agora é Prática

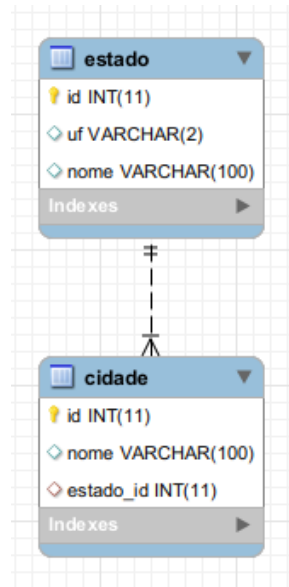
Agora é Prática

- 1) Criar banco de dados com o nome **cxacademy**
- 2) Criar tabelas Estado e Cidade e sua relação 1:n (um-para-n) com base no modelo abaixo:

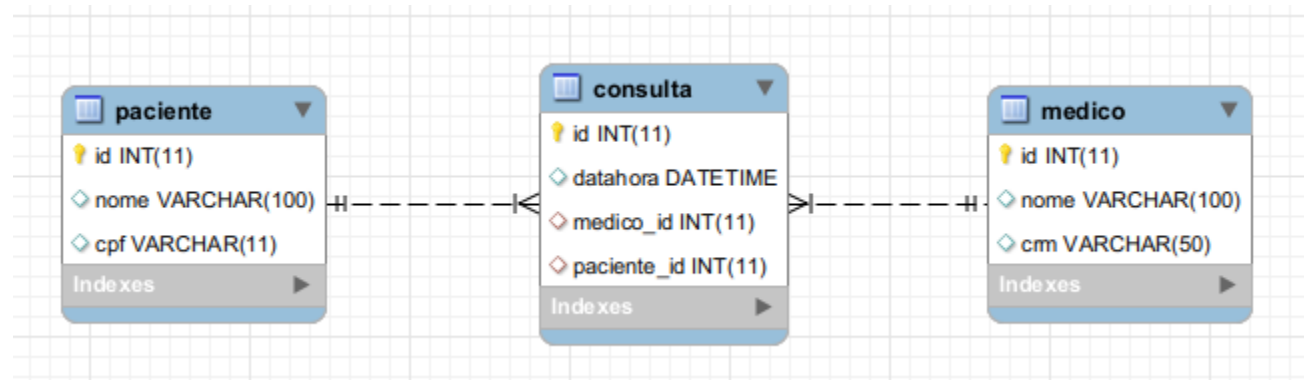


Agora é Prática

- 3) Popular as duas tabelas com alguns estados e cidades
- 4) Criar relatório com a lista de estados
- 5) Criar relatório que liste as cidades conforme o estado informado

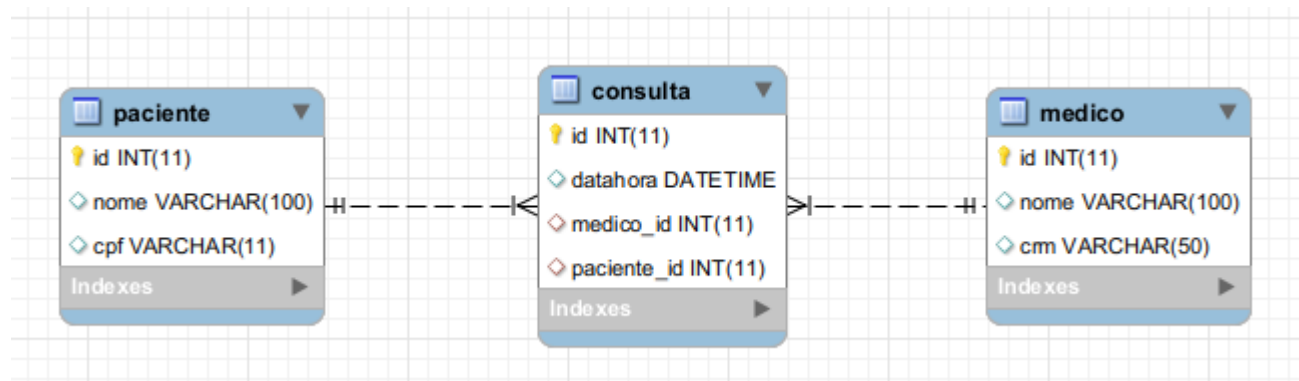


- 1) Criar tabelas Médico, Paciente e Consulta. Sendo a tabela Consulta uma tabela intermediária (associação). Ou seja, a relação entre Médico e Paciente é n:n (n-para-n) com base no modelo abaixo:



Agora é Prática

- 2) Popular as tabelas com alguns pacientes e médicos.
- 3) Criar relatório com a lista de pacientes
- 4) Criar relatório com a lista de médicos
- 5) Criar relatório com a lista de pacientes conforme o médico informado
- 6) Criar relatório com a lista de médicos conforme o paciente informado
- 7) Criar relatório com a quantidade de pacientes que cada médico possui



- Referências para estudos
- <https://www.w3schools.com/sql/>
- <https://www.devmedia.com.br/forum/qual-a-diferenca-entre-ddl-e-dml/563525>
- <https://www.devmedia.com.br/conceitos-fundamentais-de-banco-de-dados/1649>
- <https://docs.microsoft.com/pt-br/office/troubleshoot/access/database-normalization-description>
- <https://www.youtube.com/watch?v=2E7crqRI1iE>
- https://www.youtube.com/playlist?list=PLucm8g_ezqNoNHU8tjVeHmRGBFnjDI1xD
- https://www.youtube.com/playlist?list=PLucm8g_ezqNrWAQH2B_0AnrFY5dJcgOLR

