



Formação Desenvolvedor Moderno

Módulo: Banco de Dados

Capítulo: Linguagem SQL DDL e DML

Introdução

<https://devsuperior.com.br>

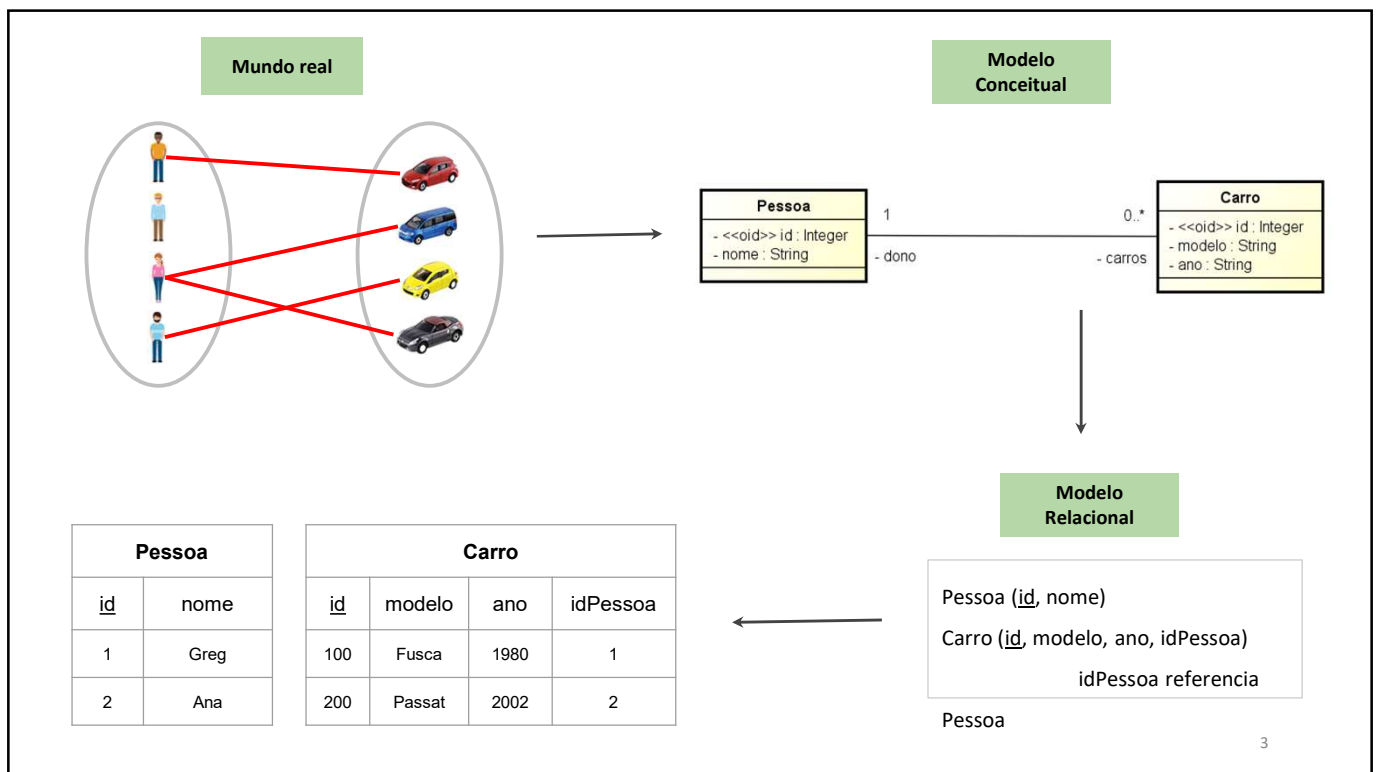
1

Agenda

- Contextualização
- Linguagem SQL
- Sublinguagens SQL

2

2



3

O que é SQL?

- *Structured Query Language*
 - Linguagem de Consulta Estruturada
- Linguagem padrão para acesso a banco de dados
- Origem:
 - Linguagem SEQUEL IBM, em 1973
- Padrão ANSI

4

4

SQL - Sublinguagens

Pode ser dividida em:

- **Data Definition Language (DDL)**
 - Comandos que definem a estrutura ou esquema do banco de dados
 - Principais comandos:
 - **CREATE** - Criação de banco de dados/tabelas/visões etc.
 - **ALTER** - Alteração da estrutura do banco de dados/tabelas/visões
 - **DROP** - Deleta um banco de dados/tabelas/visões

5

5

SQL - Sublinguagens

- **Data Manipulation Language (DML)**
 - Comandos que fazem o gerenciamento dos dados da base de dados
 - Principais comandos:
 - **SELECT** - Selecionar dados em uma ou mais tabelas/visões
 - **INSERT** - Insere dados em uma tabela/visão
 - **UPDATE** - Atualiza dados de uma tabela/visão
 - **DELETE** - Apaga dados de uma tabela/visão
 - **CALL** - Permite fazer a chamada de um procedimento

6

6

SQL - Sublinguagens

- **Data Control Language (DCL)**
 - Comandos que permitem o controle de permissão no BD
 - Principais comandos:
 - **GRANT** - Concessão de privilégios de acesso
 - **REVOKE** - Retirada de privilégios de acesso

7

7

SQL - Sublinguagens

- **Transaction Control (TCL)**
 - Comandos usados para gerenciar as mudanças feitas por comandos DML.
 - Permitem que comandos sejam agrupados em transações lógicas
 - **COMMIT** - Confirma as alterações realizadas pela transação
 - **SAVEPOINT** - Cria pontos de controle
 - **ROLLBACK** - Desfaz as alterações realizadas pela transação

8

8

Banco de Dados Postgresql

- Sistema de Gerenciamento de Banco de Dados Relacional
- Gratuito e código aberto
- Utiliza o padrão SQL

9

9

Data Definition Language (DDL)

10

Linguagem DDL

Comando **CREATE**

```
CREATE TABLE table_name (  
    column1 datatype column_constraint,  
    column2 datatype column_constraint,  
    table_constraints  
);
```

Para mais informações, buscar por **CREATE table statement postgres** para ver a estrutura completa do comando

11

11

Linguagem DDL

Tipos de dados do Postgres - Tipos Texto

Tipo	Tamanho	Observação
CHAR(n)	Fixo	até 255 caracteres. Utiliza os n bytes, mesmo que o tamanho do campo seja inferior.
VARCHAR(n)	Variável	até 65535 caracteres. utiliza 1 byte para guardar o tamanho e ocupa somente o tamanho do campo
TEXT	Variável	até 65535 bytes

12

12

Linguagem DDL

Diferença entre *CHAR* e *VARCHAR*

Valor	CHAR(4)	Exigência Armazenamento	VARCHAR(4)	Exigência Armazenamento
' '	' '	4 bytes	' '	1 byte
'ab'	'ab '	4 bytes	'ab'	3 bytes
'abcd'	'abcd'	4 bytes	'abcd'	5 bytes
'abcdefgh'	'abcd'	4 bytes	'abcd'	5 bytes

13

13

Linguagem DDL

Tipos de dados do Postgres - Tipos Numéricos

Tipo	Tamanho	Observação
SMALLINT	2 bytes	-32768 a 32767 normal. 0 a 65535 UNSIGNED.
INTEGER	4 bytes	-2147483648 a 2147483647 normal. 0 a 4294967295 UNSIGNED
SERIAL	4 bytes	Similar ao Integer. Usado para gerar uma sequência de inteiros que são frequentemente usados como a chave primária de uma tabela
FLOAT	8 bytes	Números com ponto flutuantes

14

14

Linguagem DDL

Tipos de dados do Postgres - Tipos Data e Hora

Tipo	Tamanho	Observação
DATE	4 bytes	Formato : YYYY-MM-DD
TIME	8 bytes	Formato: HH:MI:SS
TIMESTAMP	8 bytes	Formato: YYYY-MM-DD HH:MI:SS

15

15

Linguagem DDL

Restrições de Integridade - Atributos

- Restrição NOT NULL
 - Colunas de preenchimento obrigatório
 - Proíbe que o atributo receba valor NULL
- Cláusula DEFAULT
 - associa um valor default para um atributo, caso nenhum outro valor seja especificado

```
CREATE TABLE cliente (  
    nome varchar(20) NOT  
    NULL,  
    saldo int DEFAULT 0
```

16

16

Linguagem DDL

Restrições de Integridade - **Atributos**

- Restrição CHECK
 - Especifica um predicado que precisa ser satisfeito por todas as tuplas de uma relação.

```
CREATE TABLE cliente (  
    saldo int CHECK(saldo >= 0)  
);
```

17

17

Linguagem DDL

Restrições de Integridade - **Chave**

- Cláusula PRIMARY KEY
 - Identifica os atributos da relação que formam sua chave primária
 - Os atributos são implicitamente NOT NULL
 - Há duas formas de se definir a chave primária

18

18

Linguagem DDL

Restrições de Integridade - **Chave**

- Cláusula PRIMARY KEY
 - Identifica os atributos da relação que formam sua chave primária
 - Os atributos são implicitamente NOT NULL
 - Há duas formas de se definir a chave primária

Forma 1

```
CREATE TABLE cliente (  
    id serial PRIMARY KEY  
);
```

19

19

Linguagem DDL

Restrições de Integridade - **Chave**

- Cláusula PRIMARY KEY
 - Identifica os atributos da relação que formam sua chave primária
 - Os atributos são implicitamente NOT NULL
 - Há duas formas de se definir a chave primária

Forma 1

```
CREATE TABLE cliente (  
    id serial PRIMARY KEY  
);
```

Forma 2

```
CREATE TABLE cliente (  
    id serial,  
    PRIMARY KEY (id)  
);
```

20

20

Linguagem DDL

Restrições de Integridade - **Chave**

- Cláusula UNIQUE
 - Não permite valores duplicados para um determinado atributo

```
CREATE TABLE cliente (  
    id int PRIMARY KEY,  
    rg VARCHAR(12) UNIQUE  
);
```

21

21

Linguagem DDL

Restrições de Integridade - **Chave**

- Cláusula FOREIGN KEY
 - Utilizada para definir uma chave estrangeira

```
CREATE TABLE cliente (  
    id int PRIMARY KEY,  
    nome varchar(20) NOT NULL  
);
```

```
CREATE TABLE fone_cliente (  
    id int,  
    fone varchar(12),  
    PRIMARY KEY(id, fone),  
    FOREIGN KEY(id) references cliente(id)  
);
```

22

22

Linguagem DDL

Propriedades da chave estrangeira

```
cliente (id, nome)

fone_cliente (id, fone)
    id referencia cliente
```

cliente	
<u>id</u>	nome
1	Maria
2	Bob
3	Isabela

fone_cliente	
<u>id</u>	fone
1	34 98991022
2	34 98712144
2	34 23241101

23

23

Linguagem DDL

Propriedades da chave estrangeira

```
cliente (id, nome)

fone_cliente (id, fone)
    id referencia cliente
```

Cenário: Remover a cliente de id = 1 (Maria)

cliente	
<u>id</u>	nome
1	Maria
2	Bob
3	Isabela

fone_cliente	
<u>id</u>	fone
1	34 98991022
2	34 98712144
2	34 23241101

24

24

Linguagem DDL

Propriedades da chave estrangeira

- **RESTRICT**
 - **Não permite** que um registro seja deletado caso alguma tupla esteja referenciando.
- **CASCADE**
 - Deleta o registro e todas as tuplas que referenciam o registro

Importante: *Por padrão, o banco de dados aplica a cláusula RESTRICT caso nenhuma seja especificada*

25

25

Linguagem DDL

Propriedades da chave estrangeira

Sintaxe RESTRICT

```
CREATE TABLE cliente (  
    id int PRIMARY KEY,  
    nome varchar(20) NOT NULL  
);
```

```
CREATE TABLE fone_cliente (  
    id int,  
    fone varchar(12),  
    PRIMARY KEY(id, fone),  
    FOREIGN KEY(id) references cliente(id)  
    ON DELETE RESTRICT  
);
```

26

26

Linguagem DDL

Propriedades da chave estrangeira

Exemplo RESTRICT

cliente	
id	nome
1	Maria
2	Bob
3	Isabela

fone_cliente	
id	fone
1	34 98991022
2	34 98712144
2	34 23241101

Não permite que o registro de id = 1 na tabela de cliente seja deletado, pois viola a restrição de integridade

27

27

Linguagem DDL

Propriedades da chave estrangeira

Sintaxe CASCADE

```
CREATE TABLE cliente (  
    id int PRIMARY KEY,  
    nome varchar(20) NOT NULL  
);
```

```
CREATE TABLE fone_cliente (  
    id int,  
    fone varchar(12),  
    PRIMARY KEY(id, fone),  
    FOREIGN KEY(id) references cliente(id)  
    ON DELETE CASCADE  
);
```

28

28

Linguagem DDL

Propriedades da chave estrangeira

Exemplo CASCADE

cliente	
<u>id</u>	nome
1	Maria
2	Bob
3	Isabela

fone_cliente	
<u>id</u>	fone
1	34 98991022
2	34 98712144
2	34 23241101

O registro de id = 1 na tabela de cliente é deletado juntamente com todas as tuplas que as referenciam.

29

29

Linguagem DDL

Propriedades da chave estrangeira

Exemplo CASCADE

cliente	
<u>id</u>	nome
2	Bob
3	Isabela

fone_cliente	
<u>id</u>	fone
2	34 98712144
2	34 23241101

Resultado após aplicar a deleção utilizando a propriedade CASCADE

30

30

Linguagem DDL

Restrições de Integridade - **Chave**

- Cláusula FOREIGN KEY
 - Utilizada para definir uma chave estrangeira

```
CREATE TABLE cliente (  
    id int PRIMARY KEY,  
    nome varchar(20) NOT NULL  
);
```

```
CREATE TABLE fone_cliente (  
    id int,  
    fone varchar(12),  
    PRIMARY KEY(id, fone),  
    FOREIGN KEY(id) references cliente(id)  
);
```

31

31

Exercício: Linguagem DDL

De acordo com o esquema a seguir, crie o banco de dados empresa e suas tabelas

```
tb_cargo (id, nome, nivel, salario)  
tb_departamento (id, nome, sigla)  
tb_funcionario (id, nome, data_adm, sexo, cod_cargo, cod_depto)  
    cod_cargo referencia tb_cargo  
    cod_depto referencia tb_departamento
```

32

32

Exercício: Linguagem DDL

Instância

tb_cargo			
<u>id</u>	nome	nivel	salario
1	Analista de Sistemas	JR	3000
2	Desenvolvedor	JR	2500

tb_departamento		
<u>id</u>	nome	sigla
1	Informática	INF
2	Financeiro	FIN

tb_funcionario					
<u>id</u>	nome	data_adm	sexo	cod_cargo	cod_depto
1	Pedro Nogueira	12/03/2021	M	1	1
2	Maria Silva	22/04/2021	F	2	1

33

33

Linguagem DDL

Comando **DROP**

```
DROP { DATABASE | TABLE } ;
```

Apaga a estrutura de um banco de dados, tabela, visão, etc.

34

34

Linguagem DDL

Comando **ALTER TABLE**

- Altera a estrutura de uma tabela

35

35

Linguagem DDL

Comando **ALTER TABLE**

- Altera a estrutura de uma tabela
 - **Adicionar** uma nova coluna

```
ALTER TABLE table_name ADD column_name datatype;
```

36

36

Linguagem DDL

Comando **ALTER TABLE**

- Altera a estrutura de uma tabela
 - **Remover** uma coluna

```
ALTER TABLE table_name DROP COLUMN column_name;
```

37

37

Linguagem DDL

Comando **ALTER TABLE**

- Altera a estrutura de uma tabela
 - **Alterar o nome** de uma coluna existente

```
ALTER TABLE table_name RENAME [COLUMN] column_name TO  
new_column_name;
```

38

38

Linguagem DDL

Comando **ALTER TABLE**

- Altera a estrutura de uma tabela
 - **Alterar o tipo de dado** da coluna

```
ALTER TABLE table_name ALTER [COLUMN] column_name TYPE  
new_datatype
```

39

39

Linguagem DDL

Comando **ALTER TABLE**

- Altera a estrutura de uma tabela
 - **Adicionar** uma chave primária (PK)

```
ALTER TABLE table_name ADD PRIMARY KEY(column_name) ;
```

- **Remover** uma chave primária (PK)

```
ALTER TABLE table_name DROP CONSTRAINT primary_key_constraint;
```

40

40

Linguagem DDL

Comando **ALTER TABLE**

- Altera a estrutura de uma tabela
 - **Adicionar** uma chave estrangeira (FK)

```
ALTER TABLE table_name ADD FOREIGN KEY(column_name) REFERENCES  
source_table(source_column_name)
```

- **Remover** uma chave estrangeira

```
ALTER TABLE table_name DROP CONSTRAINT foreign_key_constraint;
```

41

41

Data Manipulation Language (DML)

42

Linguagem DML

Comando **INSERT**

- Permite incluir um conjunto de registros ou tuplas em uma determinada tabela
 - Há duas formas de se incluir um registro

Forma 1

```
INSERT INTO table_name VALUES (v1, v2, ..., vN);
```

Ordem dos atributos deve ser mantida

43

43

Linguagem DML

Comando **INSERT**

- Permite incluir um conjunto de registros ou tuplas em uma determinada tabela
 - Há duas formas de se incluir um registro

Forma 2

```
INSERT INTO table_name (C1, C2, ..., Cn) VALUES (v1, v2, ..., vn);
```

Ordem dos atributos não precisa ser mantida

44

44

Exercício: Linguagem DML

De acordo com o esquema EMPRESA, faça o seed no banco de dados, de acordo com as instâncias fornecidas

```
tb_cargo (id, nome, nivel, salario)
tb_departamento (id, nome, sigla)
tb_funcionario (id, nome, data_adm, sexo, cod_cargo, cod_depto)
    cod_cargo referencia tb_cargo
    cod_depto referencia tb_departamento
```

45

45

Instâncias

tb_funcionario					
id	nome	data_adm	sexo	cod_cargo	cod_depto
1	Pedro Nogueira	12/03/2021	M	1	1
2	Maria Silva	22/04/2021	F	2	1
3	Rafaela Silveira	05/07/2020	F	2	1
4	Ricardo Oliveira	06/04/2019	M	5	2
5	Tamara Junqueira	01/11/2015	F	3	1
6	Janete Rosa	01/10/2021	F	4	3
7	Fernando Silva	02/04/2022	M	3	1
8	Rosana Vieira	07/08/2018	F	2	1
9	Leandro Chaves	13/05/2017	M	5	2
10	João Marques	15/03/2021	M	3	1

tb_departamento		
id	nome	sigla
1	Informática	INF
2	Financeiro	FIN
3	Recursos Humanos	RH

tb_cargo			
id	nome	nivel	salario
1	Analista de Sistemas	JR	3000
2	Desenvolvedor	JR	2500
3	Desenvolvedor	Pleno	5200
4	Atendente	NA	1212
5	Contador	NA	5000

46

46

Linguagem DML

Comando UPDATE

- Altera dados específicos de uma tabela

```
UPDATE table_name SET column = new_value WHERE predicate;
```

47

47

Instâncias

tb_funcionario					
id	nome	data_adm	sexo	cod_cargo	cod_depto
1	Pedro Nogueira	12/03/2021	M	1	1
2	Maria Silva	22/04/2021	F	2	1
3	Rafaela Silveira	05/07/2020	F	2	1
4	Ricardo Oliveira	06/04/2019	M	5	2
5	Tamara Junqueira	01/11/2015	F	3	1
6	Janete Rosa	01/10/2021	F	4	3
7	Fernando Silva	02/04/2022	M	3	1
8	Rosana Vieira	07/08/2018	F	2	1
9	Leandro Chaves	13/05/2017	M	5	2
10	João Marques	15/03/2021	M	3	1

tb_departamento		
id	nome	sigla
1	Informática	INF
2	Financeiro	FIN
3	Recursos Humanos	RH

tb_cargo			
id	nome	nivel	salario
1	Analista de Sistemas	JR	3000
2	Desenvolvedor	JR	2500
3	Desenvolvedor	Pleno	5200
4	Atendente	NA	1212
5	Contador	NA	5000

Atualizar o nome do funcionário de id = 4 para "Ricardo Fernandes Oliveira"

48

48

Instâncias

tb_funcionario					
id	nome	data_adm	sexo	cod_cargo	cod_depto
1	Pedro Nogueira	12/03/2021	M	1	1
2	Maria Silva	22/04/2021	F	2	1
3	Rafaela Silveira	05/07/2020	F	2	1
4	Ricardo Fernandes Oliveira	06/04/2019	M	5	2
5	Tamara Junqueira	01/11/2015	F	3	1
6	Janete Rosa	01/10/2021	F	4	3
7	Fernando Silva	02/04/2022	M	3	1
8	Rosana Vieira	07/08/2018	F	2	1
9	Leandro Chaves	13/05/2017	M	5	2
10	João Marques	15/03/2021	M	3	1

tb_departamento		
id	nome	sigla
1	Informática	INF
2	Financeiro	FIN
3	Recursos Humanos	RH

tb_cargo			
id	nome	nivel	salario
1	Analista de Sistemas	JR	3000
2	Desenvolvedor	JR	2500
3	Desenvolvedor	Pleno	5200
4	Atendente	NA	1212
5	Contador	NA	5000

49

49

Linguagem DML

Comando DELETE

- Remove dados específicos de uma tabela

```
DELETE FROM table_name WHERE predicate;
```

50

50

Instâncias

tb_funcionario					
id	nome	data_adm	sexo	cod_cargo	cod_depto
1	Pedro Nogueira	12/03/2021	M	1	1
2	Maria Silva	22/04/2021	F	2	1
3	Rafaela Silveira	05/07/2020	F	2	1
4	Ricardo Oliveira	06/04/2019	M	5	2
5	Tamara Junqueira	01/11/2015	F	3	1
6	Janete Rosa	01/10/2021	F	4	3
7	Fernando Silva	02/04/2022	M	3	1
8	Rosana Vieira	07/08/2018	F	2	1
9	Leandro Chaves	13/05/2017	M	5	2
10	João Marques	15/03/2021	M	3	1

tb_departamento		
id	nome	sigla
1	Informática	INF
2	Financeiro	FIN
3	Recursos Humanos	RH

tb_cargo			
id	nome	nivel	salario
1	Analista de Sistemas	JR	3000
2	Desenvolvedor	JR	2500
3	Desenvolvedor	Pleno	5200
4	Atendente	NA	1212
5	Contador	NA	5000

Remover o funcionário "João Marques"

51

51

Linguagem DML

Comando DELETE

- Utilizando as cláusulas RESTRICT e CASCADE

```
DELETE FROM table_name WHERE predicate;
```

52

52