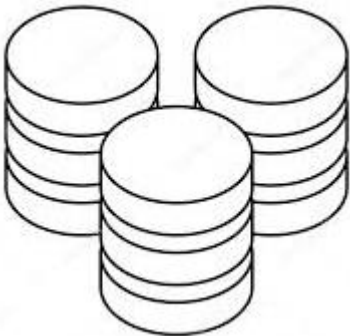


Banco de Dados II

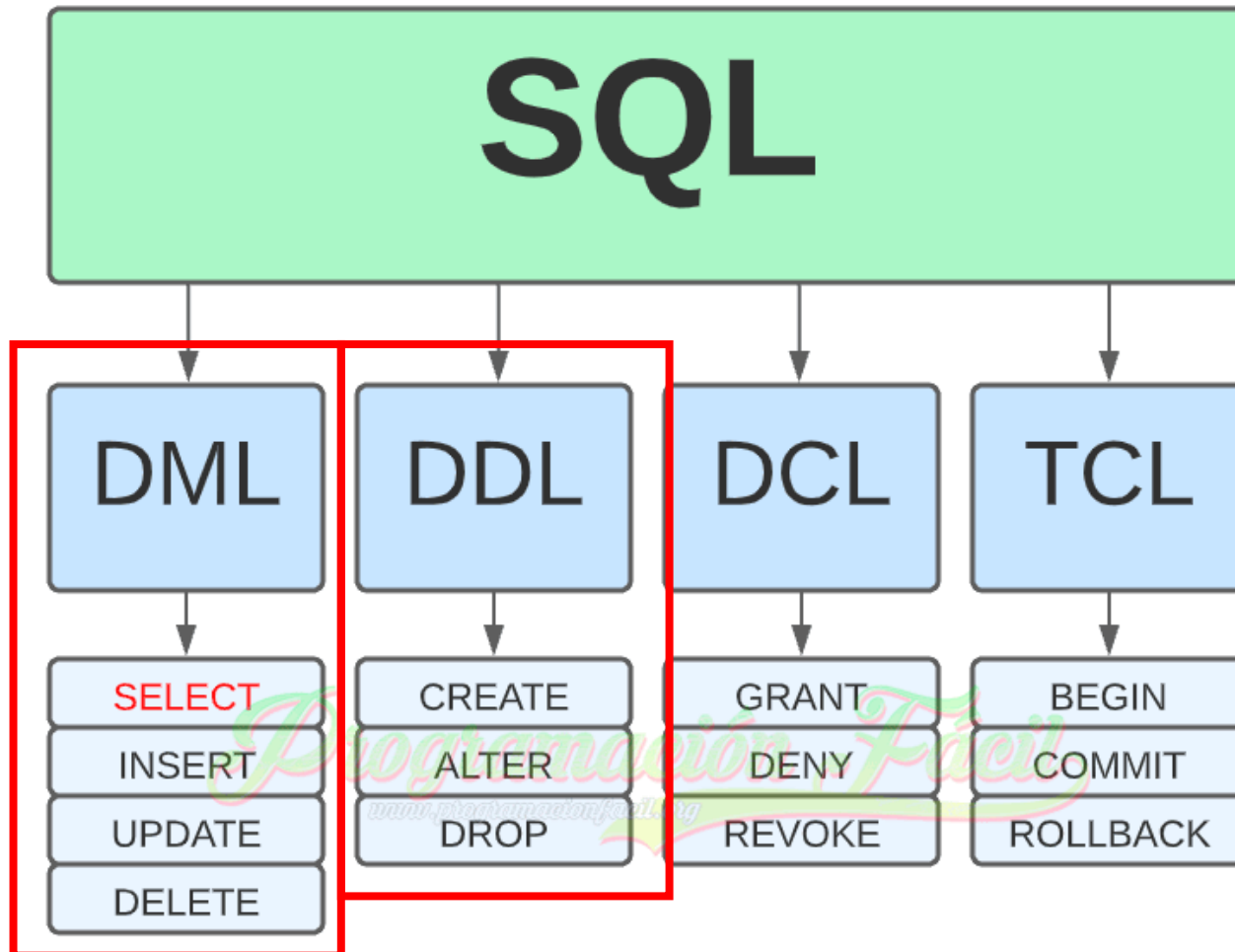
**Data Definition Language (DDL)
Data Manipulation Language (DML)
Constraints**



Profa. Damires Souza
damires@ifpb.edu.br



Linguagem SQL



Linguagem de Manipulação de Dados

```
INSERT INTO produto VALUES (1, 'Queijo', 9.99);
```

```
INSERT INTO produto (num, nome, preço)  
VALUES (1, 'Queijo', 9.99);
```

```
INSERT INTO produto (num, nome, preço)  
VALUES (1, 'Queijo', DEFAULT);
```

```
INSERT INTO produto (num, nome, preço) VALUES  
    (1, 'Queijo', 9.99),  
    (2, 'Pão', 1.99),  
    (3, 'Leite', 2.99);
```

Linguagem de Manipulação de Dados

```
UPDATE produto  
SET preço = 10  
WHERE preço = 5;
```

```
UPDATE produto  
SET preço = preço * 1.10;
```

```
UPDATE teste  
SET a = 5, b = 3, c = 1  
WHERE a > 0;
```

Linguagem de Manipulação de Dados

```
DELETE FROM produto  
WHERE preço = 10;
```

```
DELETE FROM  
produto;
```

Data Manipulation Language (DML)

Linguagem de Definição de Dados

- **O Comando Create Table:**

CREATE TABLE <tabela>

(<descrição das **colunas+tipos**>)

(<descrição das **chaves/restrições**>);

ID_EMP	NOME	SOBREN	RG	DT_NAS	ENDER	SEXO	SAL	DEPTO	ID_GER
17206-2	José	da Silva	24535357	25/02/1970	Rua Floriano Peixoto, 25	M	7000	1	
12584-7	Cecília	Ortiz Rezende	65784123	12/10/1980	Rua Marechal Deodoro da Fonseca, 335	F	3200	3	17206-2
16764-6	Pedro	Silvestre	24789635	15/05/1975	Rua Bela Cintra, 159	M	2800	4	17206-2
17987-5	Felipe	Guilhermino	25896314	19/08/1982	Rua Bolívia, 348	M	1800	3	12584-7
15698-3	Luciana	Feltosa	59874123	03/03/1980	Rua Presidente Dutra, 25	F	1500	4	16764-5
12357-1	Fabio	Santos Silva	25874137	29/07/1978	Rua Barão de Telfe, 30	M	1500	4	16764-5
16257-2	Elaine	Cristina	35789418	29/04/1974	Rua Itália, 258	F	2500	3	12584-7
15234-1	Cleiton	Fernandes	54698742	30/01/1979	Rua Andradina, 487	M	2200	3	12584-7

- **O Comando Drop Table**

DROP TABLE <tabela>;

- **O Comando Truncate Table**

TRUNCATE TABLE <tabela>;

**Data Definition
Language (DDL)**


Tipo de dados Serial

```
CREATE TABLE tablename (  
    colname SERIAL  
);
```

is equivalent to specifying:

```
CREATE SEQUENCE tablename_colname_seq;  
CREATE TABLE tablename (  
    colname integer NOT NULL DEFAULT nextval('tablename_colname_seq')  
);  
ALTER SEQUENCE tablename_colname_seq OWNED BY tablename.colname;
```

Exemplo

>  teste

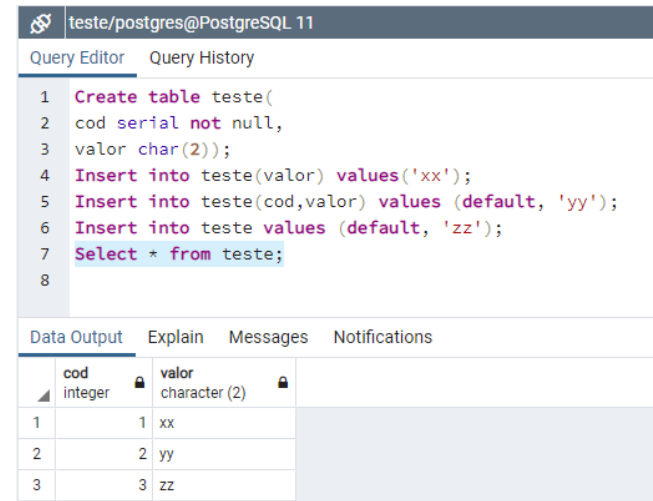
Create table **teste**(
 cod **serial** not null,
 valor char(2));

Insert into teste(valor) values('xx');

Insert into teste(cod,valor) values (default, 'yy');

Insert into teste values (default, 'zz');

Select * from teste;



The screenshot shows a PostgreSQL query editor window titled 'teste/postgres@PostgreSQL 11'. It has two tabs: 'Query Editor' and 'Query History'. The 'Query Editor' tab contains the following SQL code:

```
1 Create table teste(  
2   cod serial not null,  
3   valor char(2));  
4 Insert into teste(valor) values('xx');  
5 Insert into teste(cod,valor) values (default, 'yy');  
6 Insert into teste values (default, 'zz');  
7 Select * from teste;  
8
```

Below the query editor, there are four tabs: 'Data Output', 'Explain', 'Messages', and 'Notifications'. The 'Data Output' tab is active, displaying a table with the results of the 'Select * from teste;' query. The table has two columns: 'cod' (integer) and 'valor' (character (2)). The data is as follows:

	cod integer	valor character (2)
1	1	xx
2	2	yy
3	3	zz

ALTER TABLE

ALTER TABLE <tabela> <opção> <coluna>;

■ Principais opções:

- **ADD** - adiciona coluna
- **ADD CONSTRAINT** - adiciona restrição
- **ALTER COLUMN** - modifica definição de coluna
- **DROP CONSTRAINT** - apaga restrição
- **DROP COLUMN** - apaga coluna
- **RENAME TO** - altera o nome da tabela
- **RENAME COLUMN** - altera nome de coluna
- **RENAME CONSTRAINT** – altera nome de constraint

**Não confundir
com UPDATE**

ALTER TABLE - exemplos

ALTER TABLE Teste ADD telefone char(16);

ALTER TABLE Teste ADD fone char(16) default 'não listado';

Select * from teste;

ALTER TABLE Teste DROP COLUMN telefone;

ALTER TABLE Teste alter column valor type varchar(30);

ALTER TABLE teste RENAME TO teste2;

ALTER TABLE Teste2 RENAME COLUMN valor TO
descrição;

Select * from teste2;

ALTER TABLE - exemplos

public

> Collations

> Domains

> FTS Configurations

> FTS Dictionaries

> FTS Parsers

> FTS Templates

> Foreign Tables

> Functions

> Materialized Views

> Procedures

> Sequences

> Tables (1)

teste2

General

Columns

Indexes

Constraints

Parameters

Security

SQL

Inherited from table(s)

Select to inherit from...

Columns

+

		Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?
		cod	integer			<div>Yes</div>	<div>No</div>
		descrição	character varying	30		<div>No</div>	<div>No</div>
		fone	character	16		<div>No</div>	<div>No</div>

Data Output

Explain

Messages

Noti

Teste

Data Output

Explain

Messages

No

Teste2

	cod	integer	valor	character (2)	telefone	character (16)
1	1	xx		[null]		
2	2	yy		[null]		
3	3	zz		[null]		

	cod	integer	descrição	character varying (30)
1	1	xx		
2	2	yy		
3	3	zz		

Restrições (constraints)

- Uma **restrição** ou **constraint** é um mecanismo que implementa **controle** para garantir a **consistência** dos dados

Modelo
Relacional

Nome	Uso
NULL	Informa se o campo em questão pode ser nulo; caso contrário deve ser precedido de NOT.
UNIQUE	Indica que os valores na coluna ou conjunto de colunas não podem ser repetidos;
PRIMARY KEY	Identifica a chave primária da tabela;
FOREIGN KEY	Identifica uma chave estrangeira da tabela; implementada pela cláusula <i>references</i> .
CHECK	Determina uma regra de validação.

Constraint

[constraint <nome>] primary key | unique | not null | foreign key

- A constraint pode ser **nomeada**
 - Caso não seja, o SGBD relacional cria um nome internamente
 - Exemplo:
 - create table EMP (
EMPNO number **constraint pk_emp** primary
key, . . .);

Opções

```
CREATE TABLE Produto (  
    CodPROD integer PRIMARY KEY,  
    Descricao Varchar(20),  
    Valor Numeric(10,2),  
    Unidade Char(2));
```

VS

```
CREATE TABLE Produto (  
    CodPROD integer,  
    Descricao Varchar(20),  
    Valor Numeric(10,2),  
    Unidade Char(2),  
    Constraint pk_prod PRIMARY KEY(codProd));
```

Exemplo

**** Criar banco de dados “acadêmico”**

```
CREATE TABLE Aluno
( MatricAlu      serial NOT NULL,
  NomeAlu       VARCHAR(40),
  DataAniver    DATE default '01/01/1998',
  Sexo          CHAR(1),
  CONSTRAINT PKAluno PRIMARY KEY(MatricAlu));
```

Select * from aluno;

ALTER TABLE para *Constraints*

ALTER TABLE <tabela> <opção> <coluna>;

- ADD CONSTRAINT - adiciona restrição
- MODIFY CONSTRAINT - modifica definição de restrição
- DROP CONSTRAINT - apaga restrição
- RENAME CONSTRAINT - troca o nome da restrição

▪ **ALTER TABLE professor DROP PRIMARY KEY;**

ou

▪ **ALTER TABLE professor DROP CONSTRAINT pkprof;**

Exemplo

```
CREATE TABLE Disciplina (  
    CodDISC    serial NOT NULL,  
    NomeDISC   VARCHAR(30),  
    CONSTRAINT PKDisciplina PRIMARY KEY(CodDISC));
```

Select * from disciplina;

Exemplo

```
CREATE TABLE Professor (  
    MatricProf    serial    NOT NULL,  
    NomeProf      VARCHAR(40),  
    DataAdmissao  DATE      );
```

```
ALTER TABLE Professor ADD CONSTRAINT  
PKProfessor PRIMARY KEY(MatricProf);
```

Select * from professor;

Integridade Referencial

Tabela: **Disciplina**

**Relacionamento
entre Disciplina
e Professor**

CodDisc	NomeDisc
1	Banco de Dados II
2	Sistemas Operacionais
3	POO

Tabela: **Professor**

Matric	Nome	DataAdm	CodDisc
1	Crishane	12/12/08	1
2	Damires	12/12/09	1
3	Luciana	12/12/10	2
4	Fausto	12/12/07	3

Integridade Referencial

- Garante a não corrupção dos dados, de modo a não permitir um registro "filho" sem um registro "pai"
- Regras possíveis:
 - **RESTRICT/NO ACTION**
 - Desabilita o **delete/update** no dado referenciado (PKs não podem ser atualizadas ou excluídas)
 - **SET TO NULL:**
 - Quando o dado referenciado (PK) é **atualizado/excluído**, todos os dados dependentes associados são setados pra **NULL**.

Integridade Referencial

- **SET TO DEFAULT:**

- Quando o dado referenciado (PK) é **atualizado/excluído**, todos os dados dependentes associados são setados pra um valor **DEFAULT**.

- **CASCADE:**

- Quando o dado referenciado (PK) é **atualizado**, todos os dados dependentes associados são **atualizados**
- Quando uma linha referenciada (PK) é **deletada**, todas as linhas dependentes associadas são **deletadas**.

Vamos testar...

1. NO ACTION (default)

- As chaves especificadas **não** podem ser atualizadas ou excluídas

2. DELETE SET NULL

- Indica que, quando uma linha que contém PK for excluída, todas as linhas (FK) da tabela filha terão seu **valor ajustado para null**

3. DELETE CASCADE

- Quando uma linha contendo chave especificada for excluída, **todas as linhas da tabela filha também serão;**

4. UPDATE CASCADE

- Quando uma linha contendo chave especificada for atualizada, **todas as linhas da tabela filha serão.**

Exemplo – Integridade Referencial nº 1

ALTER TABLE professor add codDisc integer;

ALTER TABLE professor ADD CONSTRAINT FKprofdisc
FOREIGN KEY(codDisc) REFERENCES Disciplina;

Insert into disciplina(nomeDisc) values ('BDII');

insert into disciplina(nomeDisc) values ('Sistemas
Operacionais');

Select * from disciplina;

Exemplo – Integridade Referencial nº 1

```
Insert into professor(nomeProf, dataadmissao,coddisc)  
values('Crishane', null, 1);
```

```
Insert into professor(nomeProf, dataadmissao,coddisc)  
values('Damires', null, 1);
```

```
Insert into professor(nomeProf, dataadmissao,coddisc)  
values('Luciana', null, 2);
```

```
select * from professor;
```

```
Delete from disciplina where codDisc = 1;
```

O que aconteceu???


```
75 Delete from disciplina where codDisc = 1;
```

```
76
```

Data Output Explain Messages Notifications

ERROR: update or delete on table "disciplina" violates foreign key constraint "fkprofdisc" on table "professor"
DETAIL: Key (coddisc)=(1) is still referenced from table "professor".
SQL state: 23503




Exemplo – Integridade Referencial nº 2






```
Alter table professor drop constraint fkprofdisc;  
ALTER TABLE professor ADD CONSTRAINT  
FK2_prof_disc FOREIGN KEY(codDisc)  
REFERENCES Disciplina on delete set null;
```

```
Select * from disciplina;  
Select * from professor;
```

```
Delete from disciplina where  
codDisc = 1;
```

O que aconteceu???

Data Output	Explain	Messages	Notific
 coddisc [PK] integer 	nomedisc character varying (30) 		
1	2	Sistemas Operacionais	

Data Output	Explain	Messages	Notifications
 matricprof [PK] integer 	nomeprof character varying (40) 	dataadmissao date 	coddisc integer 
1	3	Luciana	[null] 2
2	1	Crishane	[null]
3	2	Damires	[null]

Exemplo – Integridade Referencial nº 3

```
Alter table professor drop constraint fk2_prof_disc;  
ALTER TABLE professor ADD CONSTRAINT  
FK3_prof_disc FOREIGN KEY(coddisc)  
REFERENCES Disciplina ON DELETE CASCADE;
```

```
Select * from disciplina;
```

```
Select * from professor;
```

--atualizações dos dados

Exemplo – Integridade Referencial nº 3

--atualizações dos dados

Insert into disciplina(coddisc,nomeDisc) values (1,'BDII');

Update professor

set coddisc = 1

where matricprof = 1 or matricprof = 2;

Select * from disciplina;

Select * from professor;

**Delete from disciplina where
codDisc = 1;**

O que aconteceu???

Data Output		Explain	Messages	Notific
	coddisc [PK] integer		nomedisc character varying (30)	
1		2	Sistemas Operacionais	

Data Output		Explain	Messages	Notifications	
	matricprof [PK] integer		nomeprof character varying (40)	dataadmissao date	coddisc integer
1		3	Luciana	[null]	2

Exemplo – Integridade Referencial nº 4

--Inserir dados

```
Insert into disciplina(coddisc,nomeDisc) values (1,'BDII');
```

```
Insert into professor(nomeProf, dataadmissao,coddisc)  
values('Crishane', null, 1);
```

```
Insert into professor(nomeProf, dataadmissao,coddisc)  
values('Damires', null, 1);
```

```
Alter table professor drop constraint fk3_prof_disc;  
ALTER TABLE professor ADD CONSTRAINT  
FK4_prof_disc FOREIGN KEY(coddisc)  
REFERENCES Disciplina ON UPDATE CASCADE;
```

Exemplo – Integridade Referencial nº 4

Select * from disciplina;

Select * from professor;

Update disciplina

Set coddisc = 11

where coddisc = 1;

O que aconteceu???

Exemplo – Integridade Referencial nº 4

Data Output		Explain	Messages	Notifi
	coddisc [PK] integer		nomedisc character varying (30)	
1		2	Sistemas Operacionais	
2		11	BDII	

Data Output		Explain	Messages	Notifications	
	matricprof [PK] integer		nomeprof character varying (40)	dataadmissao date	coddisc integer
1		3	Luciana	[null]	2
2		4	Crishane	[null]	11
3		5	Damires	[null]	11

Voltando os dados ao original



Update disciplina





Set coddisc = 1

Where coddisc = 11;

select * from disciplina;

select * from professor;

coddisc 	nomedisc 
[PK] integer	character varying (30)
2	Sistemas Operacionais
1	BDII

idprof 	nomeprof 	dataadmissao 	coddisc 
[PK] integer	character varying (40)	date	integer
3	Luciana	[null]	2
4	Crishane	[null]	1
5	Damires	[null]	1

Eliminando tabela com dependências

DROP TABLE disciplina cascade constraints;



Check Constraint

- Uma **check constraint** permite a especificação de condições sobre cada linha da tabela, para **validação de dados**
 - Deve-se referir a colunas naquela tabela
- Uma **check constraint** vai ser definida através dos comandos **CREATE TABLE** ou **ALTER TABLE**

```
CONSTRAINT constraint_name CHECK  
(column_name condition) [DISABLE]
```

Exemplo

```
CREATE TABLE nota_aluno  
  (Matricalu integer,  
   NumNota integer,  
   Nota numeric(3,1) CHECK (nota > 0),  
   constraint pk_alu_nota primary key(Matricalu,Numnota),  
   constraint FK_alu foreign Key(Matricalu) references Aluno);
```

```
Select * from aluno;
```

```
Select * from nota_aluno;
```

Exemplo

Insert into aluno values(default,'Jonas Melo','12/02/2000','M');

Insert into nota_aluno values(1,1,10);

Insert into nota_aluno values(1,2,0);

O que ocorreu?

Exemplo

matricalu [PK] integer	nomealu character varying (40)	dataaniver date	sexo character (1)
1	Jonas Melo	2000-02-12	M

matricalu [PK] integer	numnota [PK] integer	nota numeric (3,1)
1	1	10.0

Data Output Explain Messages Notifications

ERROR: new row for relation "nota_aluno" violates check constraint "nota_aluno_nota_check"
DETAIL: Failing row contains (1, 2, 0.0).
SQL state: 23514

Exemplo

```
ALTER TABLE professor ADD salario  
numeric(15,2) CHECK (salario >= 4000.00 and  
salario <= 20000.00);
```

**** Altere a tabela, insira ou atualize registros e verifique.**

O que ocorreu?