

# AULA 10 – SQL: DML

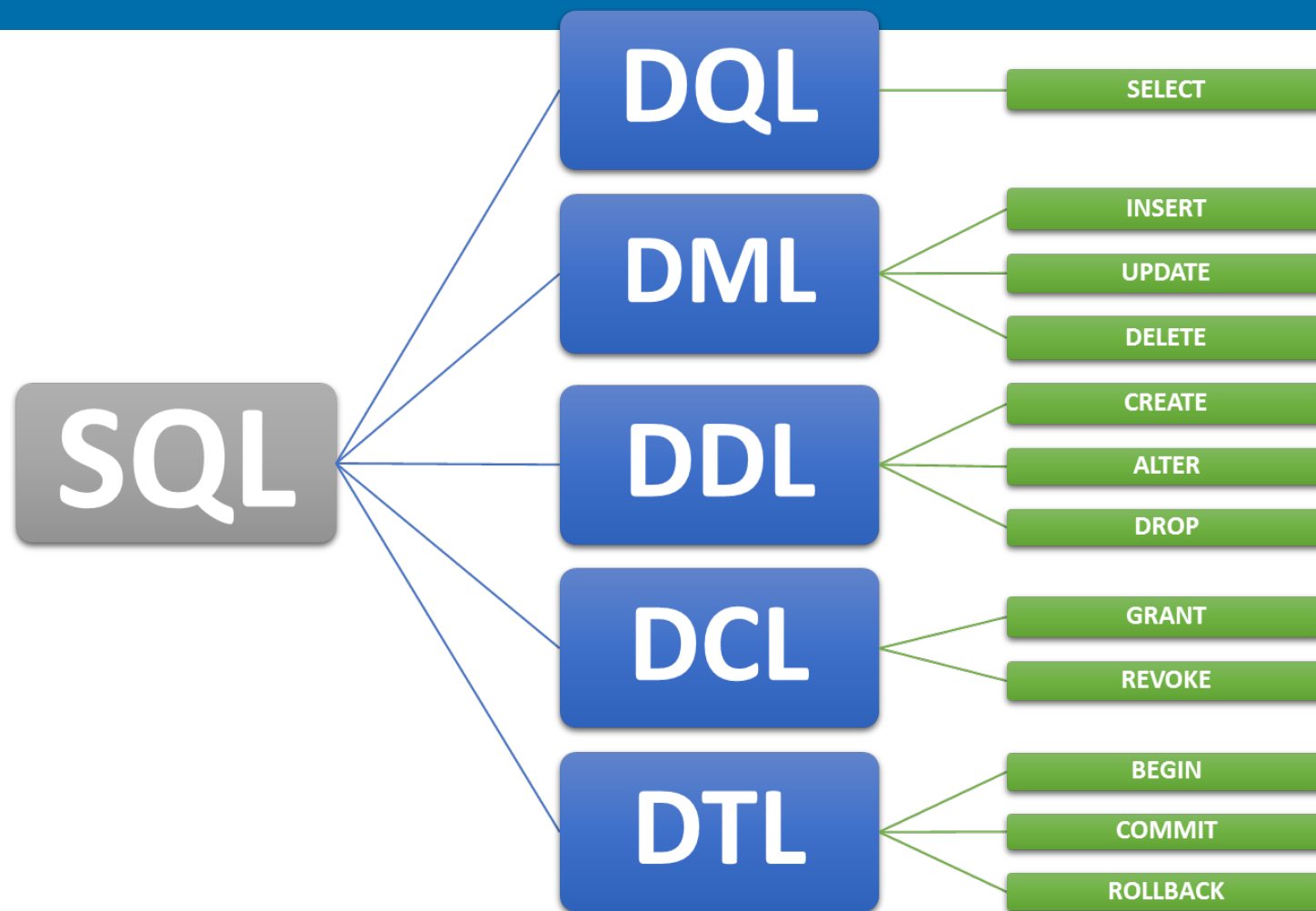
PROFA. DRA. LEILA BERGAMASCO

CC5232 – Banco de Dados

# NA AULA DE HOJE

- DML

# ORGANIZAÇÃO DO SQL



# DML

- Define o comando utilizado para que possamos manipular (INSERT, DELETE, UPDATE) os dados armazenados no banco de dados.
- INSERT: Insere registros na tabela

```
INSERT INTO <tabela> (atributo1, atributo2,..., atributo n)  
VALUES (V1,V2,...,VN)
```

- professor(cpf-prof,nome-prof,sobrenome-prof,salario)

```
CREATE TABLE professor (  
    cpf varchar,  
    nome-prof varchar,  
    sobrenome-prof varcahr  
    salario numeric DEFAULT 9.99  
);
```

# DML

```
CREATE TABLE professor (  
    cpf varchar,  
    nome_prof varchar,  
    sobrenome_prof varchar,  
    salario numeric DEFAULT 9.99  
);
```

```
INSERT INTO professor VALUES  
('3532222-22', 'Leila', 'Bergamasco', 500);
```

```
INSERT INTO professor (cpf,nome-prof,sobrenome-prof) VALUES  
('3532222-22', 'Leila', 'Bergamasco');
```

varchar


int

Utilizando default

# DML

Auto-increment

```
CREATE TABLE professor (  
    id serial NOT NULL,  
    nome_prof varchar,  
    sobrenome_prof varchar,  
    salario numeric DEFAULT 9.99,  
    PRIMARY KEY (id)  
);
```



```
INSERT INTO professor (id,nome_prof,sobrenome_prof,salario) VALUES (DEFAULT, 'Leila', 'Bergamasco',500);
```

```
INSERT INTO professor (nome_prof,sobrenome_prof,salario)  
VALUES ('Leila', 'Bergamasco',500);
```

```
INSERT INTO professor (nome_prof,sobrenome_prof)  
VALUES ('Leila', 'Bergamasco');
```

# DML

- Define o comando utilizado para que possamos manipular (INSERT, DELETE, UPDATE) os dados armazenados no banco de dados.
- DELETE: Remove registros na tabela

```
DELETE FROM tabela  
WHERE predicado;
```

- professor(cpf-prof, nome-prof, sobrenome-prof, salario)

```
CREATE TABLE professor (  
    id numeric,  
    nome-prof varchar,  
    sobrenome-prof varcahr  
    salario numeric DEFAULT 9.99  
);
```

# DML

```
DELETE FROM professor  
WHERE id=01
```

- DELETE e chave estrangeira!



# DML

```
CREATE TABLE sala (  
    id_sala numeric NOT NULL,  
    nro sala varchar  
    PRIMARY KEY (id_sala)  
);
```

```
CREATE TABLE prof_sala (  
    Id_prof numeric NOT NULL,  
    Id_sala numeric NOT NULL  
);
```

```
ALTER TABLE public.prof_sala    ADD CONSTRAINT sala_fk FOREIGN KEY (id_sala)    REFERENCES public.sala  
(id_sala)    ON DELETE CASCADE
```

```
ALTER TABLE public.prof_sala    ADD CONSTRAINT prof_fk FOREIGN KEY (id_prof)    REFERENCES public.professor  
(id)    ON DELETE CASCADE
```

# DML

```
testdb=# SELECT * FROM process;
emp_id | SECTION
-----+-----
      1 | distribution
      2 | curing
      3 | technology
```

```
estdb=# SELECT * FROM employee;
```

emp_id	name	STATUS	phone_num	process_fk
1	joemarie	regular	0985959905	1
2	shakhira	probationary	093948889487	2
3	hyle	regular	095599093490	1
4	kobe	probationary	097867556451	3
5	nasty	regular	094458909099	2
6	arianne	regular	097746890988	2

```
SELECT * FROM employee;
```

emp_id	name	STATUS	phone_num	process_fk
2	shakhira	probationary	093948889487	2
4	kobe	probationary	097867556451	3
5	nasty	regular	094458909099	2
6	arianne	regular	097746890988	2

**DELETE e chave estrangeira!**

**DELETE FROM process**

**WHERE section='distribution';**

# DML

`DELETE FROM professor`

?????????



# DML

- Define o comando utilizado para que possamos manipular (INSERT, DELETE, UPDATE) os dados armazenados no banco de dados.
- UPDATE: Atualiza registros na tabela

```
UPDATE <tabela>  
SET atributo = xxxx  
WHERE predicado;
```

- Professor(cpf-prof,nome-prof,sobrenome-prof,salario)

```
CREATE TABLE professor (  
    cpf varchar,  
    nome-prof varchar,  
    sobrenome-prof varcahr  
    salario numeric DEFAULT 9.99  
);
```

# DML

```
UPDATE professor  
SET salario = 600  
WHERE nome_prof = 'Leila';
```

```
UPDATE professor  
SET salario = 600  
WHERE id = 3;
```

<https://extendsclass.com/postgresql-online.html#>

[https://www.w3schools.com/sql/trysql.asp?filename=trysql\\_op\\_in](https://www.w3schools.com/sql/trysql.asp?filename=trysql_op_in)

Problemas com essa cláusula?

# DQL

- Define o comando utilizado para que possamos consultar (SELECT) os dados armazenados no banco.
  - SELECT: registros
- Formato de uma consulta genérica

```
SELECT A1, A2, ..., An  
FROM r1, r2, ..., rm  
WHERE P
```

```
SELECT *  
FROM professor  
WHERE id=3
```

$A_i$  representa um atributo  
 $R_i$  representa uma relação  
 $P$  é um predicado

O resultado da consulta é uma nova relação!

# EXERCÍCIOS

Considere os seguintes esquemas de relação:

- Professor (prof-numero, prof-nome, prof-rua, prof-cidade, prof-telefone)
- Aluno (alu-numero, alu-nome, alu-rua, alu-cidade)
- Disciplina (disc-codigo, disc-nome, disc-quant-aulas-semana)
- Matricula(alu-numero, disc-codigo, ano, semestre, nota, frequencia)
- ProfessorDisciplina (prof-numero, disc-codigo)

- Escreva a consulta SQL considerando os comandos DDL, DQL e DML (create, select, insert, update, delete)
  - Realize as seguintes inserções:
    - 3 professores, tendo um prof-numero = 13, um sendo da cidade de São Paulo e um morando em outra cidade.
    - 3 disciplinas, sendo uma com disc-código = CC5232 e disc-Nome = Banco de Dados, sendo uma com quant-aulas-semana >3
    - 3 alunos sendo que um deles chama Bruno e uma tenha o alu-numero = 6547
  - Insira para o professor de código 13, a disciplina de código CC5232
  - Insira para o aluno de número 6547, a disciplina de código CC5232, ano 2020, semestre 2, nota 7, frequência 90
  - Remova todas as disciplinas que têm mais de 3 aulas por semana.
  - Remova todos os alunos
  - Atualize a cidade de algum professor que não mora em São Paulo, para São Paulo
  - Atualize uma das disciplinas para 2 vezes por semana

OBRIGADO E ATÉ A PRÓXIMA AULA!