

SQL: DQL

Vanessa Braganholo

vanessa@ic.uff.br

SQL

- Considerada um dos principais motivos para o sucesso dos bancos de dados relacionais comerciais
- **Structured Query Language** (Linguagem de Consulta Estruturada)
- **Declarativa**
- Instruções para definição de dados (DDL), consultas (DQL) e atualizações (DML)



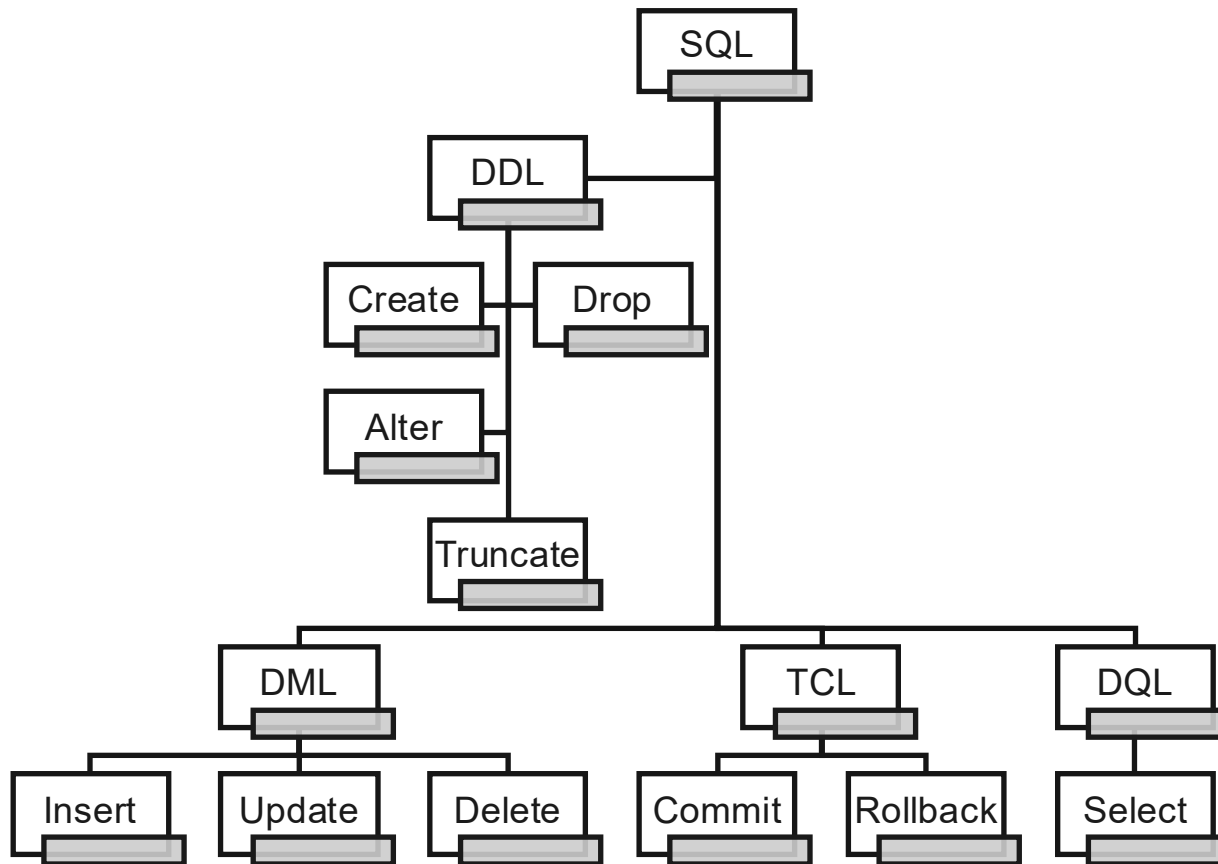
Donald Chamberlin



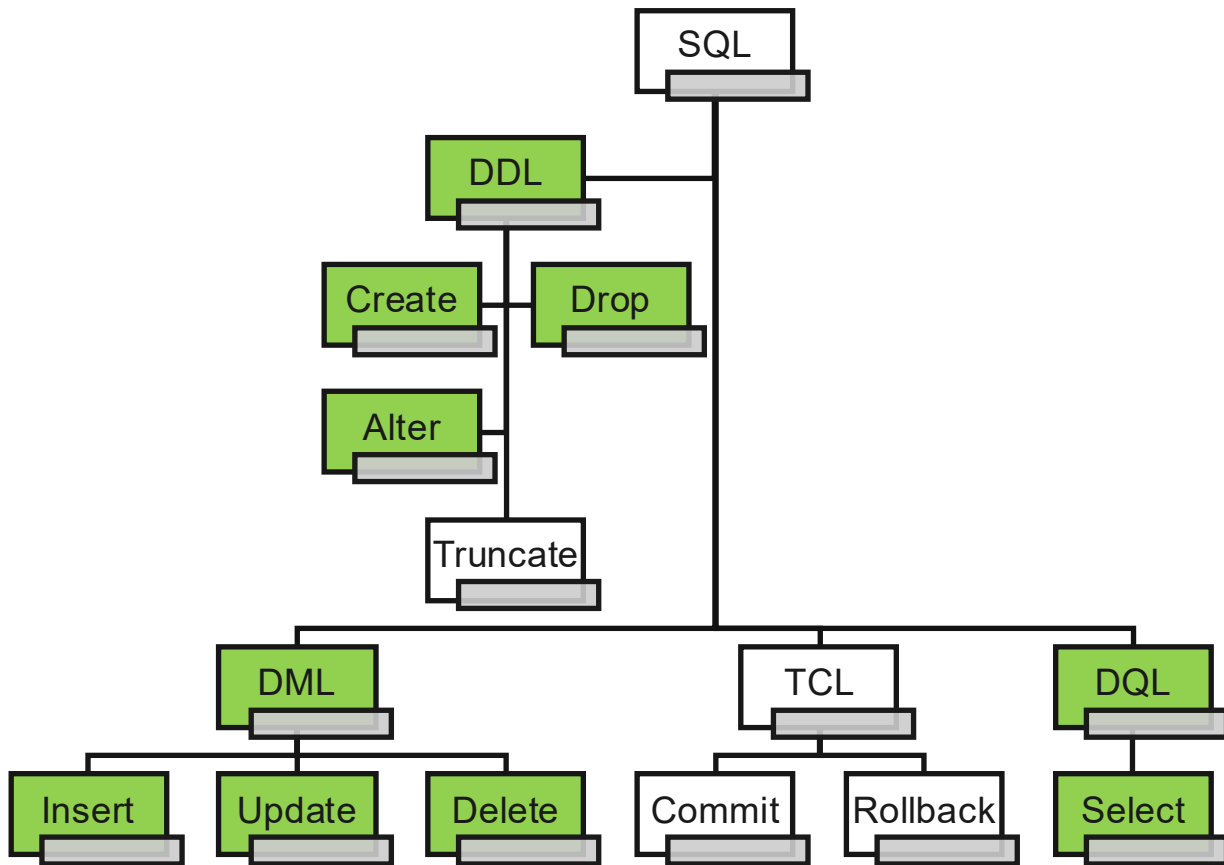
Raymond F. Boyce



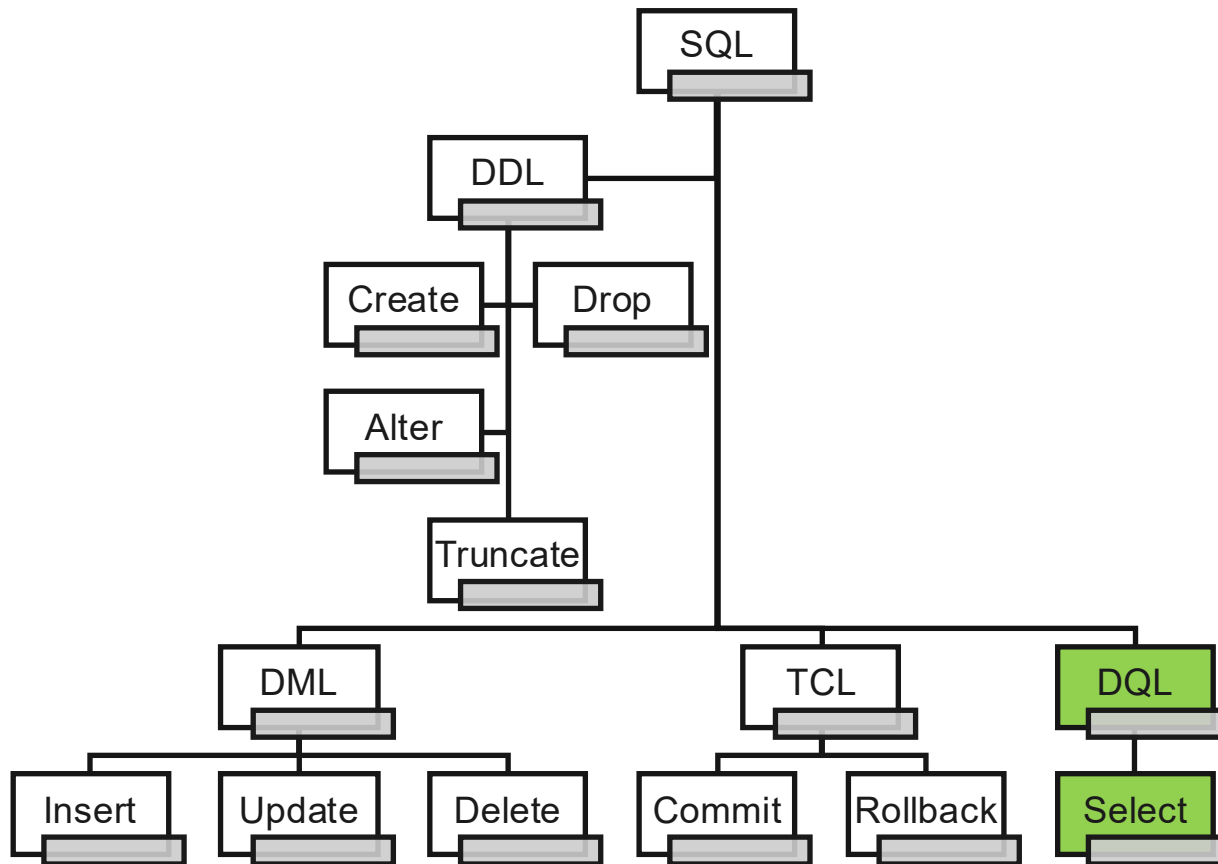
SQL - Organização



Na Disciplina



Aula de Hoje



DQL

ou “consultando meus dados”

Esquema usado nos exemplos

```
CREATE TABLE DEPARTAMENTO (
  COD_DEPTO INT NOT NULL,
  NOME VARCHAR(15) NOT NULL,
  CPF_GERENTE VARCHAR(11),
  DATA_INICIO_GERENTE DATE,
  PRIMARY KEY (COD_DEPTO),
  UNIQUE (NOME),
  FOREIGN KEY (CPF_GERENTE) REFERENCES EMPREGADO(CPF));
```

```
CREATE TABLE EMPREGADO (
  NOME VARCHAR(15) NOT NULL,
  INICIALM CHAR,
  SOBRENOME VARCHAR(15) NOT NULL,
  CPF VARCHAR(11) NOT NULL,
  DNASCIMENTO DATE,
  ENDERECO VARCHAR(30),
  SEXO CHAR,
  SALARIO FLOAT,
  CPF_SUPERVISOR VARCHAR(11),
  COD_DEPTO INT NOT NULL,
  PRIMARY KEY (CPF),
  FOREIGN KEY (CPF_SUPERVISOR) REFERENCES EMPREGADO (CPF),
  FOREIGN KEY (COD_DEPTO) REFERENCES DEPARTAMENTO (COD_DEPTO));
```

```
CREATE TABLE DEPTO_LOCALIZACOES (
  COD_DEPTO INT NOT NULL,
  LOCAL VARCHAR(15) NOT NULL,
  PRIMARY KEY (COD_DEPTO),
  FOREIGN KEY (COD_DEPTO) REFERENCES DEPARTAMENTO (COD_DEPTO));
```

```
CREATE TABLE PROJETO (
  COD_PROJ INT NOT NULL,
  NOME VARCHAR(30) NOT NULL,
  LOCAL VARCHAR(15),
  COD_DEPTO INT NOT NULL,
  PRIMARY KEY (COD_PROJ),
  UNIQUE (NOME),
  FOREIGN KEY (COD_DEPTO) REFERENCES DEPARTAMENTO (COD_DEPTO));
```

```
CREATE TABLE TRABALHA_EM (
  CPF VARCHAR(11) NOT NULL,
  COD_PROJ INT NOT NULL,
  HORAS FLOAT NOT NULL,
  PRIMARY KEY (CPF, COD_PROJ),
  FOREIGN KEY (CPF) REFERENCES EMPREGADO (CPF),
  FOREIGN KEY (COD_PROJ) REFERENCES PROJETO (COD_PROJ));
```

```
CREATE TABLE DEPENDENTE (
  EMP_CPF VARCHAR(11) NOT NULL,
  NOME VARCHAR(15) NOT NULL,
  SOBRENOME VARCHAR(15) NOT NULL,
  SEXO CHAR,
  DNASCIMENTO DATE,
  PARENTESCO VARCHAR(8),
  PRIMARY KEY (EMP_CPF, NOME),
  FOREIGN KEY (EMP_CPF) REFERENCES EMPREGADO(CPF));
```

EMPREGADO

Í	NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO
	John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	5
	Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000	88866...	5
	Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	5
	Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	5
	James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	1
	Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	4
	Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4
	Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4

DEPARTAMENTO

COD_DEPTO	NOME	CPF_GERENTE	DATA_INICIO_GERENTE
1	Sede administrativa	88866555500	1981-06-19
4	Administracao	98765432100	1995-01-01
5	Pesquisa	33344555500	1988-05-22

DEPTO_LOCALIZACOES

COD_DEPTO	LOCAL
1	Houston
4	Staffor
5	Bellaire

PROJETO

COD_PROJ	NOME	LOCAL	COD_DEPTO
1	ProdutoX	Bellaire	5
2	ProdutoY	Sugarland	5
3	ProdutoZ	Houston	5
10	Automatizacao	Stafford	4
20	Reorganizacao	Houston	1
30	Novos Beneficios	Stafford	4

TRABALHA_EM

CPF	COD_PROJ	HORAS
12345678900	1	32.5
12345678900	2	7.5
33344555500	2	10
33344555500	3	10
33344555500	10	10
33344555500	20	10
45345345300	1	20
45345345300	2	20
66688444400	3	40
88866555500	20	40
98765432100	20	15
98765432100	30	20
98798798700	10	35
98798798700	30	5
99988777700	10	10
99988777700	30	30

DEPENDENTE

EMP_CPF	NOME	SOBRENOME	SEXO	DNASCIMENTO	PARENTESCO
12345678900	Alice	Smith	F	1988-12-30	FILHA
12345678900	Elizabeth	Smith	F	1967-05-05	CONJUGE
12345678900	Michael	Smith	M	1988-01-04	FILHO
33344555500	Alice	Braga	F	1986-04-05	FILHA
33344555500	Joy	Silva	F	1958-05-03	CONJUGE
33344555500	Theodore	Silva	M	1983-10-25	FILHO
98765432100	Abner	Soares	M	1942-02-28	CONJUGE

SQL permite que uma tabela tenha duas ou mais tuplas que são idênticas em todos os seus valores de atributo

- Multiconjunto ou comportamento *bag*
- Isso pode ser impedido se for definida uma **chave primária** para a tabela

SELECT

Instrução **SELECT**

- Uma instrução básica para recuperar informações de um banco de dados

SELECT, FROM

SELECT <Lista de Atributos>
FROM <Lista de Tabelas>

Atributos listados no **SELECT**

- Colunas que devem ser retornadas pela consulta
- Essa operação é chamada de **PROJEÇÃO**

Tabelas listadas no **FROM**

- Tabelas de onde os dados devem ser retornados
- As colunas listadas no SELECT devem pertencer às tabelas listadas no FROM

Exemplo

Selecionar todos os nomes dos empregados

```
SELECT NOME  
FROM EMPREGADO
```

Exemplo

```
SELECT NOME  
FROM EMPREGADO
```

⌵	NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO
	John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	5
	Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000	88866...	5
	Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	5
	Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	5
	James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	1
	Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	4
	Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4
	Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4

Exemplo

```
SELECT NOME  
FROM EMPREGADO
```

PROJEÇÃO da coluna NOME

NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO
John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	5
Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000	88866...	5
Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	5
Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	5
James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	1
Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	4
Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4
Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4

Exemplo com várias colunas

```
SELECT CPF, NOME, SOBRENOME  
FROM EMPREGADO
```

⌵	NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO
	John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	5
	Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000	88866...	5
	Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	5
	Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	5
	James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	1
	Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	4
	Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4
	Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4

Exemplo com várias colunas

PROJEÇÃO das colunas CPF, NOME, SOBRENOME

```
SELECT CPF, NOME, SOBRENOME
FROM EMPREGADO
```

NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO
John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	5
Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000	88866...	5
Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	5
Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	5
James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	1
Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	4
Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4
Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4

Exemplo com várias colunas

```
SELECT CPF, NOME, SOBRENOME  
FROM EMPREGADO
```

CPF	NOME	SOBRENOME
12345678900	John	Smith
33344555500	Franklin	Wong
45345345300	Joyce	English
66688444400	Ramesh	Narayan
88866555500	James	Borg
98765432100	Jennifer	Wallace
98798798700	Ahmad	Jabbar
99988777700	Alicia	Zelaya

As colunas na resposta aparecem na mesma
ordem especificada no **SELECT**

WHERE

SELECT <Lista de Atributos>
FROM <Lista de Tabelas>
WHERE <Condição>

Condição de seleção (**WHERE**)

- Condição é uma expressão booleana (ou seja, resulta VERDADEIRO OU FALSO)
- Operadores relacionais e operadores lógicos
 - =, <, <=, >, >= e <>
 - and, or, not
- Para que uma tupla **t** esteja no resultado, **a condição deve ser verdadeira para os atributos de t**
- Essa operação é chamada de **SELEÇÃO**

Exemplo WHERE

Selecionar todos os nomes dos empregados que ganham mais de 30.000

```
SELECT NOME  
FROM EMPREGADO  
WHERE SALARIO > 30000;
```

Exemplo WHERE

```
SELECT NOME  
FROM EMPREGADO  
WHERE SALARIO > 30000;
```

NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO
John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	5
Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000	88866...	5
Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	5
Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	5
James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	1
Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	4
Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4
Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4

Exemplo WHERE

```
SELECT NOME  
FROM EMPREGADO  
WHERE SALARIO > 30000;
```

⋮	NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO	
	John	B	Smith	12345...	1965-01...	731 Fon...	M	30000 F	33344...	5	F
	Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000 T	88866...	5	T
	Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000 F	33344...	5	F
	Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000 T	33344...	5	T
	James	E	Borg	88866...	1937-11...	450 Sto...	M	55000 T	NULL	1	T
	Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000 T	88866...	4	T
	Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000 F	98765...	4	F
	Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000 F	98765...	4	F

Exemplo WHERE

```
SELECT NOME  
FROM EMPREGADO  
WHERE SALARIO > 30000;
```

NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO
John	B	Smith	12345...	1965-01...	731 Fon...	M	30000 F	33344...	5 F
Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000 T	88866...	5 T
Joyce	A	English	45345...	1972-07...	5031 Ri...	F	25000 F	33344...	5 F
Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000 T	33344...	5 T
James	E	Borg	88866...	1937-11...	450 Sto...	M	55000 T	NULL	1 T
Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000 T	88866...	4 T
Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000 F	98765...	4 F
Alicia	J	Zelaya	99988...	1966-01...	3321 C...	F	25000 F	98765...	4 F

Exemplo WHERE

```
SELECT NOME  
FROM EMPREGADO  
WHERE SALARIO > 30000;
```

NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO	
John	B	Smith	12345...	1965-01...	731 Fon...	M	30000 F	33344...	5	F
Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000 T	88866...	5	T
Joyce	A	English	45345...	1972-07...	5031 Ri...	F	25000 F	33344...	5	F
Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000 T	33344...	5	T
James	E	Borg	88866...	1937-11...	450 Sto...	M	55000 T	NULL	1	T
Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000 T	88866...	4	T
Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000 F	98765...	4	F
Alicia	J	Zelaya	99988...	1966-01...	3321 C...	F	25000 F	98765...	4	F

Exemplo WHERE

```
SELECT NOME  
FROM EMPREGADO  
WHERE SALARIO > 30000;
```

NOME
Franklin
Ramesh
James
Jennifer

Exemplo WHERE com AND

Selecionar todos os nomes dos empregados que ganham mais de 30.000 e que são do sexo Feminino

```
SELECT NOME, SOBRENOME  
FROM EMPREGADO  
WHERE SALARIO > 30000  
AND SEXO = 'F';
```

Uma tupla t de EMPREGADO só vai para o resultado se a condição da cláusula WHERE for verdadeira para ela — ou seja, toda tupla que está no resultado satisfaz a condição de seleção

Tabela Verdade do AND

A	B	A and B
TRUE	TRUE	TRUE
TRUE	FALSE	FALSE
FALSE	TRUE	FALSE
FALSE	FALSE	FALSE

Exemplo WHERE com AND

```
SELECT NOME, SOBRENOME  
FROM EMPREGADO  
WHERE SALARIO > 30000 AND SEXO = 'F';
```

Í	NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO
	John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	5
	Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000	88866...	5
	Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	5
	Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	5
	James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	1
	Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	4
	Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4
	Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4

Exemplo WHERE com AND

```
SELECT NOME, SOBRENOME
FROM EMPREGADO
WHERE SALARIO > 30000 AND SEXO = 'F';
```



NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO
John	B	Smith	12345...	1965-01...	731 Fon...	M F	30000 F	33344...	5
Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000	88866...	5
Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	5
Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	5
James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	1
Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	4
Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4
Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4

F

Exemplo WHERE com AND

```
SELECT NOME, SOBRENOME
FROM EMPREGADO
WHERE SALARIO > 30000 AND SEXO = 'F';
```



NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO	
John	B	Smith	12345...	1965-01...	731 Fon...	M F	30000 F	33344...	5	F
Franklin	T	Wong	33344...	1955-12...	638 Vos...	M F	40000 T	88866...	5	F
Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	5	
Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	5	
James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	1	
Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	4	
Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4	
Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4	

Exemplo WHERE com AND

```
SELECT NOME, SOBRENOME
FROM EMPREGADO
WHERE SALARIO > 30000 AND SEXO = 'F';
```



NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO	
John	B	Smith	12345...	1965-01...	731 Fon...	M F	30000 F	33344...	5	F
Franklin	T	Wong	33344...	1955-12...	638 Vos...	M F	40000 T	88866...	5	F
Joyce	A	English	45345...	1972-07...	5631 Ri...	F T	25000 F	33344...	5	F
Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	5	
James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	1	
Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	4	
Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4	
Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4	

Exemplo WHERE com AND

```
SELECT NOME, SOBRENOME
FROM EMPREGADO
WHERE SALARIO > 30000 AND SEXO = 'F';
```

	NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO	
	John	B	Smith	12345...	1965-01...	731 Fon...	M F	30000 F	33344...	5	F
	Franklin	T	Wong	33344...	1955-12...	638 Vos...	M F	40000 T	88866...	5	F
	Joyce	A	English	45345...	1972-07...	5631 Ri...	F T	25000 F	33344...	5	F
➔	Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M F	38000 T	33344...	5	F
	James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	1	
	Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	4	
	Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4	
	Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4	

Exemplo WHERE com AND

```
SELECT NOME, SOBRENOME
FROM EMPREGADO
WHERE SALARIO > 30000 AND SEXO = 'F';
```

	NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO	
	John	B	Smith	12345...	1965-01...	731 Fon...	M F	30000 F	33344...	5	F
	Franklin	T	Wong	33344...	1955-12...	638 Vos...	M F	40000 T	88866...	5	F
	Joyce	A	English	45345...	1972-07...	5631 Ri...	F T	25000 F	33344...	5	F
	Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M F	38000 T	33344...	5	F
➔	James	E	Borg	88866...	1937-11...	450 Sto...	M F	55000 T	NULL	1	F
	Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	4	
	Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4	
	Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4	

Exemplo WHERE com AND

```
SELECT NOME, SOBRENOME
FROM EMPREGADO
WHERE SALARIO > 30000 AND SEXO = 'F';
```

NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO	
John	B	Smith	12345...	1965-01...	731 Fon...	M F	30000 F	33344...	5	F
Franklin	T	Wong	33344...	1955-12...	638 Vos...	M F	40000 T	88866...	5	F
Joyce	A	English	45345...	1972-07...	5631 Ri...	F T	25000 F	33344...	5	F
Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M F	38000 T	33344...	5	F
James	E	Borg	88866...	1937-11...	450 Sto...	M F	55000 T	NULL	1	F
Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F T	43000 T	88866...	4	T
Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4	
Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4	



Exemplo WHERE com AND

```
SELECT NOME, SOBRENOME
FROM EMPREGADO
WHERE SALARIO > 30000 AND SEXO = 'F';
```

NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO	
John	B	Smith	12345...	1965-01...	731 Fon...	M F	30000 F	33344...	5	F
Franklin	T	Wong	33344...	1955-12...	638 Vos...	M F	40000 T	88866...	5	F
Joyce	A	English	45345...	1972-07...	5631 Ri...	F T	25000 F	33344...	5	F
Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M F	38000 T	33344...	5	F
James	E	Borg	88866...	1937-11...	450 Sto...	M F	55000 T	NULL	1	F
Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F T	43000 T	88866...	4	T
Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M F	25000 F	98765...	4	F
Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4	



Exemplo WHERE com AND

```
SELECT NOME, SOBRENOME
FROM EMPREGADO
WHERE SALARIO > 30000 AND SEXO = 'F';
```

Í	NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO	
	John	B	Smith	12345...	1965-01...	731 Fon...	M F	30000 F	33344...	5	F
	Franklin	T	Wong	33344...	1955-12...	638 Vos...	M F	40000 T	88866...	5	F
	Joyce	A	English	45345...	1972-07...	5631 Ri...	F T	25000 F	33344...	5	F
	Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M F	38000 T	33344...	5	F
	James	E	Borg	88866...	1937-11...	450 Sto...	M F	55000 T	NULL	1	F
	Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F T	43000 T	88866...	4	T
	Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M F	25000 F	98765...	4	F
	Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F T	25000 F	98765...	4	F



Exemplo WHERE com AND

```
SELECT NOME, SOBRENOME  
FROM EMPREGADO  
WHERE SALARIO > 30000 AND SEXO = 'F';
```

NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO
John	B	Smith	12345...	1965-01...	731 Fon...	M F	30000 F	33344...	5 F
Franklin	T	Wong	33344	1955-12	638 Vos	M F	40000 T	88866	5 F
Joyce	A	English	45345	1972-07	5631 Pi	F T	25000 F	33344	5 F
Rameeh	K	Narayan	66688...	1962-08...	075 Fire...	M F	38000 T	33344...	5 F
James	E	Borg	88866...	1937-11...	450 Ste...	M F	55000 T	NULL	1 F
Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F T	43000 T	88866...	4 T
Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M F	25000 F	98765...	4 F
Alicia	J	Zelava	99988...	1968-01...	3321 C...	F T	25000 F	98765...	4 F

Exemplo WHERE com AND

```
SELECT NOME, SOBRENOME  
FROM EMPREGADO  
WHERE SALARIO > 30000 AND SEXO = 'F';
```

: NOME	SOBRENOME
Jennifer	Wallace

Exemplo WHERE com OR

Selecionar todos os nomes dos projetos executados em Houston ou pelo departamento de Código 5

```
SELECT NOME  
FROM PROJETO  
WHERE LOCAL = 'Houston'  
OR COD_DEPTO = 5;
```

Tabela Verdade do OR

A	B	A or B
TRUE	TRUE	TRUE
TRUE	FALSE	TRUE
FALSE	TRUE	TRUE
FALSE	FALSE	FALSE

Exemplo WHERE com OR

```
SELECT NOME  
FROM PROJETO  
WHERE LOCAL = 'Houston' OR COD_DEPTO = 5;
```

COD_PROJ	NOME	LOCAL	COD_DEPTO
1	ProdutoX	Bellaire	5
2	ProdutoY	Sugarland	5
3	ProdutoZ	Houston	5
10	Automatizacao	Stafford	4
20	Reorganizacao	Houston	1
30	Novos Beneficios	Stafford	4

Exemplo WHERE com OR

```
SELECT NOME  
FROM PROJETO  
WHERE LOCAL = 'Houston' OR COD_DEPTO = 5;
```



COD_PROJ	NOME	LOCAL	COD_DEPTO	
1	ProdutoX	Bellaire	5	T
2	ProdutoY	Sugarland	5	
3	ProdutoZ	Houston	5	
10	Automatizacao	Stafford	4	
20	Reorganizacao	Houston	1	
30	Novos Beneficios	Stafford	4	

Exemplo WHERE com OR

```
SELECT NOME
FROM PROJETO
WHERE LOCAL = 'Houston' OR COD_DEPTO = 5;
```

COD_PROJ	NOME	LOCAL	COD_DEPTO	
1	ProdutoX	Bellaire F	5 T	T
2	ProdutoY	Sugarland F	5 T	T
3	ProdutoZ	Houston	5	
10	Automatizacao	Stafford	4	
20	Reorganizacao	Houston	1	
30	Novos Beneficios	Stafford	4	



Exemplo WHERE com OR

```
SELECT NOME
FROM PROJETO
WHERE LOCAL = 'Houston' OR COD_DEPTO = 5;
```

COD_PROJ	NOME	LOCAL	COD_DEPTO	
1	ProdutoX	Bellaire F	5 T	T
2	ProdutoY	Sugarland F	5 T	T
3	ProdutoZ	Houston T	5 T	T
10	Automatizacao	Stafford	4	
20	Reorganizacao	Houston	1	
30	Novos Beneficios	Stafford	4	



Exemplo WHERE com OR

```
SELECT NOME
FROM PROJETO
WHERE LOCAL = 'Houston' OR COD_DEPTO = 5;
```

COD_PROJ	NOME	LOCAL	COD_DEPTO	
1	ProdutoX	Bellaire F	5 T	T
2	ProdutoY	Sugarland F	5 T	T
3	ProdutoZ	Houston T	5 T	T
10	Automatizacao	Stafford F	4 F	F
20	Reorganizacao	Houston	1	
30	Novos Beneficios	Stafford	4	



Exemplo WHERE com OR

```
SELECT NOME
FROM PROJETO
WHERE LOCAL = 'Houston' OR COD_DEPTO = 5;
```

COD_PROJ	NOME	LOCAL	COD_DEPTO	
1	ProdutoX	Bellaire F	5 T	T
2	ProdutoY	Sugarland F	5 T	T
3	ProdutoZ	Houston T	5 T	T
10	Automatizacao	Stafford F	4 F	F
20	Reorganizacao	Houston T	1 F	T
30	Novos Beneficios	Stafford	4	



Exemplo WHERE com OR

```
SELECT NOME
FROM PROJETO
WHERE LOCAL = 'Houston' OR COD_DEPTO = 5;
```

COD_PROJ	NOME	LOCAL	COD_DEPTO	
1	ProdutoX	Bellaire F	5 T	T
2	ProdutoY	Sugarland F	5 T	T
3	ProdutoZ	Houston T	5 T	T
10	Automatizacao	Stafford F	4 F	F
20	Reorganizacao	Houston T	1 F	T
30	Novos Beneficios	Stafford F	4 F	F



Exemplo WHERE com OR

```
SELECT NOME  
FROM PROJETO  
WHERE LOCAL = 'Houston' OR COD_DEPTO = 5;
```

COD_PROJ	NOME	LOCAL	COD_DEPTO	
1	ProdutoX	Bellaire F	5 T	T
2	ProdutoY	Sugarland F	5 T	T
3	ProdutoZ	Houston T	5 T	T
10	Automatizacao	Stafford F	4 F	F
20	Reorganizacao	Houston T	1 F	T
30	Novos Beneficios	Stafford F	4 F	F

Exemplo WHERE com OR

```
SELECT NOME  
FROM PROJETO  
WHERE LOCAL = 'Houston' OR COD_DEPTO = 5;
```

⋮ NOME
ProdutoX
ProdutoY
ProdutoZ
Reorganizacao

Exemplo 2: WHERE com OR

Selecionar todos os nomes dos projetos executados em Houston e Sugarland

```
SELECT NOME  
FROM PROJETO  
WHERE LOCAL = 'Houston'  
OR LOCAL = 'Sugarland';
```

Exemplo 2: WHERE com OR

Selecionar todos os nomes dos projetos executados em Houston e Sugarland

```
SELECT NOME  
FROM PROJETO  
WHERE LOCAL = 'Houston'  
OR LOCAL = 'Sugarland';
```

Porquê OR e não AND
??????

Exemplo 2: WHERE com OR

Selecionar todos os nomes dos projetos executados em Houston **e** Sugarland

```
SELECT NOME  
FROM PROJETO  
WHERE LOCAL = 'Houston'  
OR LOCAL = 'Sugarland';
```

Porquê OR e não AND
??????

Uma mesma tupla não pode
ter local Houston e Sugarland
ao mesmo tempo – a resposta
da consulta com AND seria
vazia

Exemplo 2: WHERE com OR

```
SELECT NOME
FROM PROJETO
WHERE LOCAL = 'Houston'
OR LOCAL = 'Sugarland';
```

Uma mesma tupla não pode ter local
 Houston e Sugarland ao mesmo tempo
 – a resposta da consulta com AND seria
 vazia

COD_PROJ	NOME	LOCAL	COD_DEPTO
1	ProdutoX	Bellaire	5
2	ProdutoY	Sugarland	5
3	ProdutoZ	Houston	5
10	Automatizacao	Stafford	4
20	Reorganizacao	Houston	1
30	Novos Beneficios	Stafford	4

Exemplo 2: WHERE com OR

```
SELECT NOME  
FROM PROJETO  
WHERE LOCAL = 'Houston'  
OR LOCAL = 'Sugarland';
```

COD_PROJ	NOME	LOCAL	COD_DEPTO
1	ProdutoX	Bellaire	5
2	ProdutoY	Sugarland	5
3	ProdutoZ	Houston	5
10	Automatizacao	Stafford	4
20	Reorganizacao	Houston	1
30	Novos Beneficios	Stafford	4

Exemplo 2: WHERE com OR

```
SELECT NOME  
FROM PROJETO  
WHERE LOCAL = 'Houston'  
OR LOCAL = 'Sugarland';
```

⋮ NOME
ProdutoY
ProdutoZ
Reorganizacao

Exemplo com NOT

Selecionar todos os nomes dos projetos que não foram executados em Houston

```
SELECT NOME  
FROM PROJETO  
WHERE NOT (LOCAL = 'Houston')
```

Tabela Verdade do NOT

A	not(A)
TRUE	FALSE
FALSE	TRUE

Exemplo com NOT

```
SELECT NOME  
FROM PROJETO  
WHERE NOT (LOCAL = 'Houston')
```

COD_PROJ	NOME	LOCAL	COD_DEPTO
1	ProdutoX	Bellaire	5
2	ProdutoY	Sugarland	5
3	ProdutoZ	Houston	5
10	Automatizacao	Stafford	4
20	Reorganizacao	Houston	1
30	Novos Beneficios	Stafford	4

Exemplo com NOT

```
SELECT NOME
FROM PROJETO
WHERE NOT (LOCAL = 'Houston')
```



COD_PROJ	NOME	LOCAL	COD_DEPTO
1	ProdutoX	Bellaire F	5
2	ProdutoY	Sugarland	5
3	ProdutoZ	Houston	5
10	Automatizacao	Stafford	4
20	Reorganizacao	Houston	1
30	Novos Beneficios	Stafford	4

T

Exemplo com NOT

```
SELECT NOME
FROM PROJETO
WHERE NOT (LOCAL = 'Houston')
```

COD_PROJ	NOME	LOCAL	COD_DEPTO
1	ProdutoX	Bellaire F	5
2	ProdutoY	Sugarland F	5
3	ProdutoZ	Houston	5
10	Automatizacao	Stafford	4
20	Reorganizacao	Houston	1
30	Novos Beneficios	Stafford	4

T
T

Exemplo com NOT

```
SELECT NOME  
FROM PROJETO  
WHERE NOT (LOCAL = 'Houston')
```

COD_PROJ	NOME	LOCAL	COD_DEPTO	
1	ProdutoX	Bellaire F	5	T
2	ProdutoY	Sugarland F	5	T
3	ProdutoZ	Houston T	5	F
10	Automatizacao	Stafford	4	
20	Reorganizacao	Houston	1	
30	Novos Beneficios	Stafford	4	

Exemplo com NOT

```
SELECT NOME
FROM PROJETO
WHERE NOT (LOCAL = 'Houston')
```

COD_PROJ	NOME	LOCAL	COD_DEPTO	
1	ProdutoX	Bellaire F	5	T
2	ProdutoY	Sugarland F	5	T
3	ProdutoZ	Houston T	5	F
10	Automatizacao	Stafford F	4	T
20	Reorganizacao	Houston	1	
30	Novos Beneficios	Stafford	4	



Exemplo com NOT

```
SELECT NOME
FROM PROJETO
WHERE NOT (LOCAL = 'Houston')
```

COD_PROJ	NOME	LOCAL	COD_DEPTO	
1	ProdutoX	Bellaire F	5	T
2	ProdutoY	Sugarland F	5	T
3	ProdutoZ	Houston T	5	F
10	Automatizacao	Stafford F	4	T
20	Reorganizacao	Houston T	1	F
30	Novos Beneficios	Stafford	4	

Exemplo com NOT

```
SELECT NOME
FROM PROJETO
WHERE NOT (LOCAL = 'Houston')
```

COD_PROJ	NOME	LOCAL	COD_DEPTO	
1	ProdutoX	Bellaire F	5	T
2	ProdutoY	Sugarland F	5	T
3	ProdutoZ	Houston T	5	F
10	Automatizacao	Stafford F	4	T
20	Reorganizacao	Houston T	1	F
30	Novos Beneficios	Stafford F	4	T



Exemplo com NOT

```
SELECT NOME
FROM PROJETO
WHERE NOT (LOCAL = 'Houston')
```

COD_PROJ	NOME	LOCAL	COD_DEPTO	
1	ProdutoX	Bellaire F	5	T
2	ProdutoY	Sugarland F	5	T
3	ProdutoZ	Houston T	5	F
10	Automatizacao	Stafford F	4	T
20	Reorganizacao	Houston T	1	F
30	Novos Beneficios	Stafford F	4	T

Solução equivalente (mas processada de forma diferente)

```
SELECT NOME
FROM PROJETO
WHERE LOCAL <> 'Houston'
```

COD_PROJ	NOME	LOCAL	COD_DEPTO
1	ProdutoX	Bellaire	5
2	ProdutoY	Sugarland	5
3	ProdutoZ	Houston	5
10	Automatizacao	Stafford	4
20	Reorganizacao	Houston	1
30	Novos Beneficios	Stafford	4

Solução equivalente (mas processada de forma diferente)

```
SELECT NOME
FROM PROJETO
WHERE LOCAL <> 'Houston'
```

COD_PROJ	NOME	LOCAL	COD_DEPTO	
1	ProdutoX	Bellaire T	5	T
2	ProdutoY	Sugarland T	5	T
3	ProdutoZ	Houston F	5	F
10	Automatizacao	Stafford T	4	T
20	Reorganizacao	Houston F	1	F
30	Novos Beneficios	Stafford T	4	T

Exemplo com AND e OR

Selecionar todos os nomes dos empregados que ganham mais de 40.000 e que são do sexo Feminino, ou que ganham mais de 30.000 e são do sexo Masculino

```
SELECT NOME  
FROM EMPREGADO  
WHERE (SALARIO > 40000 AND SEXO = 'F')  
OR (SALARIO > 30000 AND SEXO = 'M')
```

Exemplo WHERE com AND

```
SELECT NOME FROM EMPREGADO
WHERE (SALARIO > 40000 AND SEXO = 'F')
OR (SALARIO > 30000 AND SEXO = 'M');
```

NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO	
John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	5	F
Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000	88866...	5	T
Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	5	F
Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	5	T
James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	1	T
Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	4	T
Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4	F
Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4	F

Exemplo WHERE com AND

```
SELECT NOME FROM EMPREGADO
WHERE (SALARIO > 40000 AND SEXO = 'F')
OR (SALARIO > 30000 AND SEXO = 'M');
```

NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO	
John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	5	F
Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000	88866...	5	T
Joyce	A	English	45345...	1972-07...	5634 Ri...	F	25000	33344...	5	F
Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	5	T
James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	1	T
Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	4	T
Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4	F
Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4	F

Valores NULL

Significados de NULL

- Valor desconhecido
- Valor indisponível ou retido
- Atributo não aplicável

Cada valor NULL individual é considerado diferente de qualquer outro valor NULL

Para lidar com valores NULL, SQL usa uma lógica de três valores:

- TRUE, FALSE e UNKNOWN

Lógica de 3 Valores: AND

A	B	A and B
TRUE	TRUE	TRUE
TRUE	FALSE	FALSE
TRUE	UNKNOWN	UNKNOWN
FALSE	TRUE	FALSE
FALSE	FALSE	FALSE
FALSE	UNKNOWN	FALSE
UNKNOWN	TRUE	UNKNOWN
UNKNOWN	FALSE	FALSE
UNKNOWN	UNKNOWN	UNKNOWN

Exemplo

```
SELECT NOME, SOBRENOME  
FROM EMPREGADO  
WHERE COD_DEPTO = 5 AND CPF_SUPERVISOR = '33344555500';
```

⋮	NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO			
	John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	T	5	T	T
	Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000	88866...	F	5	T	F
	Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	T	5	T	T
	Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	T	5	T	T
	James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	U	1	F	F
	Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	F	4	F	F
	Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	F	4	F	F
	Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	F	4	F	F

Exemplo

```
SELECT NOME, SOBRENOME  
FROM EMPREGADO  
WHERE COD_DEPTO = 5 AND CPF_SUPERVISOR = '33344555500';
```

NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO			
John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	T	5	T	T
Franklin	T	Wong	33344...	1955-12...	638 Ves...	M	40000	88866...	F	5	T	F
Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	T	5	T	T
Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	T	5	T	T
James	F	Borg	88866...	1937-11	450 Sto	M	55000	NULL	U	1	F	F
Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	F	4	F	F
Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	F	4	F	F
Alicia	J	Zelava	99988...	1968-01...	3321 C...	F	25000	98765...	F	4	F	F

Exemplo (com tabela alterada)

```
SELECT NOME, SOBRENOME
FROM EMPREGADO
WHERE COD_DEPTO = 5 AND CPF_SUPERVISOR = '33344555500';
```

Í	NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO		
	John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	T 5	T	T
	Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000	88866...	F 5	T	F
	Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	T 5	T	T
	Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	T 5	T	T
	James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	U	U	U
	Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	F 4	F	F
	Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	F 4	F	F
	Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	F 4	F	F

Exemplo (com tabela alterada)

```
SELECT NOME, SOBRENOME
FROM EMPREGADO
WHERE COD_DEPTO = 5 AND CPF_SUPERVISOR = '33344555500';
```

Í	NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO		
	John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	T 5	T	T
	Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000	88866...	F 5	T	F
	Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	T 5	T	T
	Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	T 5	T	T
	James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	U	U	U
	Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	F 4	F	F
	Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	F 4	F	F
	Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	F 4	F	F

Lógica de 3 Valores: OR

A	B	A or B
TRUE	TRUE	TRUE
TRUE	FALSE	TRUE
TRUE	UNKNOWN	TRUE
FALSE	TRUE	TRUE
FALSE	FALSE	FALSE
FALSE	UNKNOWN	UNKNOWN
UNKNOWN	TRUE	TRUE
UNKNOWN	FALSE	UNKNOWN
UNKNOWN	UNKNOWN	UNKNOWN

Lógica de 3 Valores: NOT

A	not(A)
TRUE	FALSE
FALSE	TRUE
UNKNOWN	UNKNOWN

IS NULL e IS NOT NULL

Permite consultas que verificam se o valor de um atributo é NULL

- IS NULL ou IS NOT NULL

IS NULL

Selecionar todos os nomes dos empregados que não possuem supervisor

```
SELECT NOME  
FROM EMPREGADO  
WHERE CPF_SUPERVISOR IS NULL;
```

IS NULL

```
SELECT NOME  
FROM EMPREGADO  
WHERE CPF_SUPERVISOR IS NULL;
```

NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO
John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	5
Franklin	T	Wong	33344...	1955-12...	638 Ves...	M	40000	88866...	5
Joyce	A	English	45345...	1972-07...	5031 Ri...	F	25000	33344...	5
Ramesh	K	Narayan	66688...	1982-09...	973 Fire...	M	38000	33344...	5
James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	1
Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	42000	88866...	4
Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4
Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4

IS NULL

```
SELECT NOME  
FROM EMPREGADO  
WHERE CPF_SUPERVISOR IS NULL;
```

⋮ NOME
James

IS NOT NULL

Selecionar todos os nomes dos empregados que possuem supervisor

```
SELECT NOME  
FROM EMPREGADO  
WHERE CPF_SUPERVISOR IS NOT NULL;
```

IS NOT NULL

```
SELECT NOME  
FROM EMPREGADO  
WHERE CPF_SUPERVISOR IS NOT NULL;
```

NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO
John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	5
Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000	88866...	5
Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	5
Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	5
James	F	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	1
Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	4
Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4
Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4

IS NOT NULL

```
SELECT NOME  
FROM EMPREGADO  
WHERE CPF_SUPERVISOR IS NOT NULL;
```

⋮ NOME
John
Joyce
Ramesh
Franklin
Jennifer
Ahmad
Alicia

SELECT *

Usado para recuperar todos os valores de atributo das tuplas selecionadas na consulta

Exemplo: Selecionar todos os dados dos empregados que trabalham no departamento de código 5.

```
SELECT *  
FROM EMPREGADO  
WHERE COD_DEPTO = 5;
```

SELECT *

```
SELECT *  
FROM EMPREGADO  
WHERE COD_DEPTO = 5;
```

Í	NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO
	John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	5
	Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000	88866...	5
	Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	5
	Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	5
	James	E	Borg	88866...	1937-11...	450 Sto...	M	55000	NULL	1
	Jennifer	S	Wallace	98765...	1941-06...	291 Ber...	F	43000	88866...	4
	Ahmad	V	Jabbar	98798...	1969-03...	980 Dall...	M	25000	98765...	4
	Alicia	J	Zelaya	99988...	1968-01...	3321 C...	F	25000	98765...	4

SELECT *

```
SELECT *  
FROM EMPREGADO  
WHERE COD_DEPTO = 5;
```

PROJEÇÃO de todas as
colunas da tabela resultante
da consulta

⋮	NOME	INICIA...	SOBR...	CPF	DNAS...	ENDE...	SEXO	SALA...	CPF_S...	COD_DEPTO
	John	B	Smith	12345...	1965-01...	731 Fon...	M	30000	33344...	5
	Franklin	T	Wong	33344...	1955-12...	638 Vos...	M	40000	88866...	5
	Joyce	A	English	45345...	1972-07...	5631 Ri...	F	25000	33344...	5
	Ramesh	K	Narayan	66688...	1962-09...	975 Fire...	M	38000	33344...	5

Resumo “Parcial”

```
SELECT <lista de atributos>  
FROM <lista de tabelas>  
[ WHERE <condição>];
```

Para praticar...

Vamos usar o SQLite

No Google Classroom, baixe o arquivo airport.db e salve numa pasta do seu computador

No VSCode, abra essa pasta pelo menu File / Open Folder

Clique no SQLite Explorer e depois em New Query

Escreva a consulta, clique com o botão da direita e escolha Run Query

O passo a passo completo está no material **SQLite no VSCode**

Atenção: no SQLite a comparação com = diferencia maiúsculas de minúsculas ('ES' não é igual a 'es')

```

aircrafttype (AircraftTypeID, AircraftName)
airport (AirportID, AirportCode, AirportName, CityName,
          CountryCode, NumRunways, NumTerminals)
aircraft (AircraftID, AircraftTypeID, RegNum, LastMaintEnd,
          NextMaintBegin, NextMaintEnd);
          AircraftTypeID referencia aircrafttype
route (RouteID, Origin, Destination, Distance, Duration, RouteStatus)
          Origin references airport (AirportID)
          Destination references airport (AirportID)
class (ClassID, ClassName)
flight (FlightID, RouteID, AircraftID)
          AircraftID referencia aircraft
          RouteID referencia route
flightclass (FlightID, ClassID, MaxSeats, BasePrice)
          FlightID referencia flight
          ClassID referencia class
flightdep (FlightID, DepDay, DepTime); //horários de vôos
          FlightID referencia flight
log (RecordID, ByUser, Note, EventTime)
pax (RecordID, FlightID, FlightDate, ClassID, PaxName, PaxRef, Note)
          FlightID referencia flight
          ClassID referencia class
stats (FlightID, FlightDate, ClassID, CurrSeats, CurrPrice)
          FlightID referencia flight
          ClassID referencia class

```

Exercícios

1. Faça uma consulta que retorne todos os aeroportos da cidade de London.
2. Faça uma consulta que retorne os aeroportos que possuem mais de 1 terminal.
3. Faça uma consulta que retorne o número máximo de assentos (MaxSeats) do voo cujo FlightID é 876 na classe cujo ID é 3 (usar a tabela flightclass).
4. Faça uma consulta que retorna o ID das rotas que possuem distância maior do que 5000 milhas.
5. Faça uma consulta que retorna os nomes dos passageiros (PaxName da tabela pax) que estão no voo (FlightID) 652 e Class ID 3.
6. Faça uma consulta que retorna o código e nome dos aeroportos da Espanha (countryCode = 'ES').
7. Faça uma consulta que retorna os dias (DepDay) e horários (DepTime) de partida do voo 896.
8. Faça uma consulta que retorna a data da próxima manutenção (NextMaintBegin) da aeronave cujo número de registro é ZX5731.
9. Faça uma consulta que retorna o número de registro das aeronaves cuja última manutenção terminou (LastMaintEnd) entre 2007-04-01 e 2008-05-20 (inclusive).
10. Faça uma consulta que retorna o número de registro das aeronaves do tipo (AircraftTypeID) 617 e 503

SQLite Online

Se não conseguir instalar a extensão no VSCode, você também pode usar o <https://sqliteonline.com/>



Perguntas?

Vanessa Braganholo

vanessa@ic.uff.br