

Banco de Dados

Jose.wellington@ceub.edu.br

Planejamento

1 Dia

a. Manhã

Modelagem de dados

b. Tarde

Normalização de dados

2 Dia

c. Manhã

Comandos DDL

1 avaliação – Modelagem de dados e Normalização

d. Tarde

Comandos DML – DQL Avançados

3 Dia

e. Manhã

Banco de Dados NoSQL – Redis

2 avaliação – Comandos SQL

f. Tarde

NoSQL – MongoDB

NoSQL – Neo4j

Agenda

- **SGBD - Sistema Gerenciador de Banco de Dados**
- **Representação Física do SGBD**
- **SQL - Structured Query Language**
- **Categorias do SQL**
- **Exercício prático**

SGBD

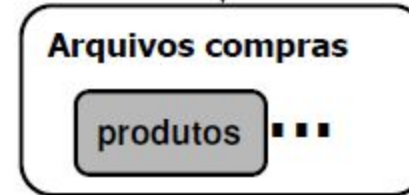
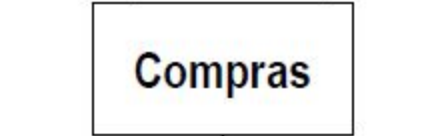
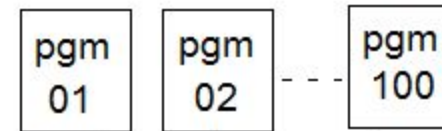
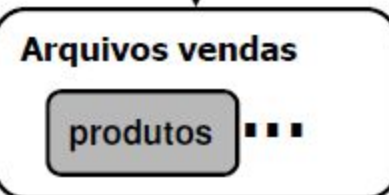
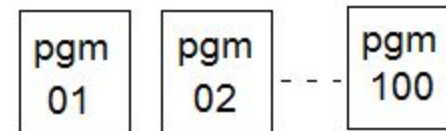
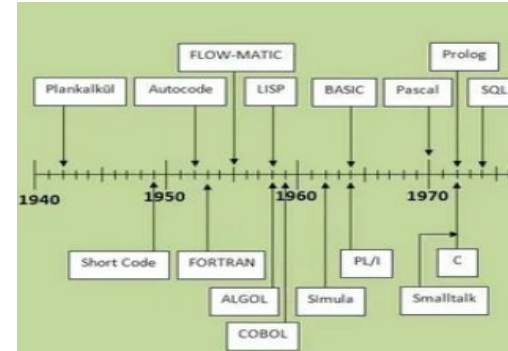
Sistema Gerenciador de Banco de Dados

Processamento sem BD

Dados não Compartilha dados

Tb Funcionário		
Código	Nome	Cd cargo
001	João	01
002	Maria	02
003	Patricia	05

Ta Cargo	
Cd cargo	Cargo
01	Gerente
02	Vendedor
03	Contador



Processamento com BD

Exemplo:

Coleção de Dados Relacionados



Sistema de Vendas

Sistema de Produção

Sistema de Engenharia

Banco de Dados

Dados de Produtos

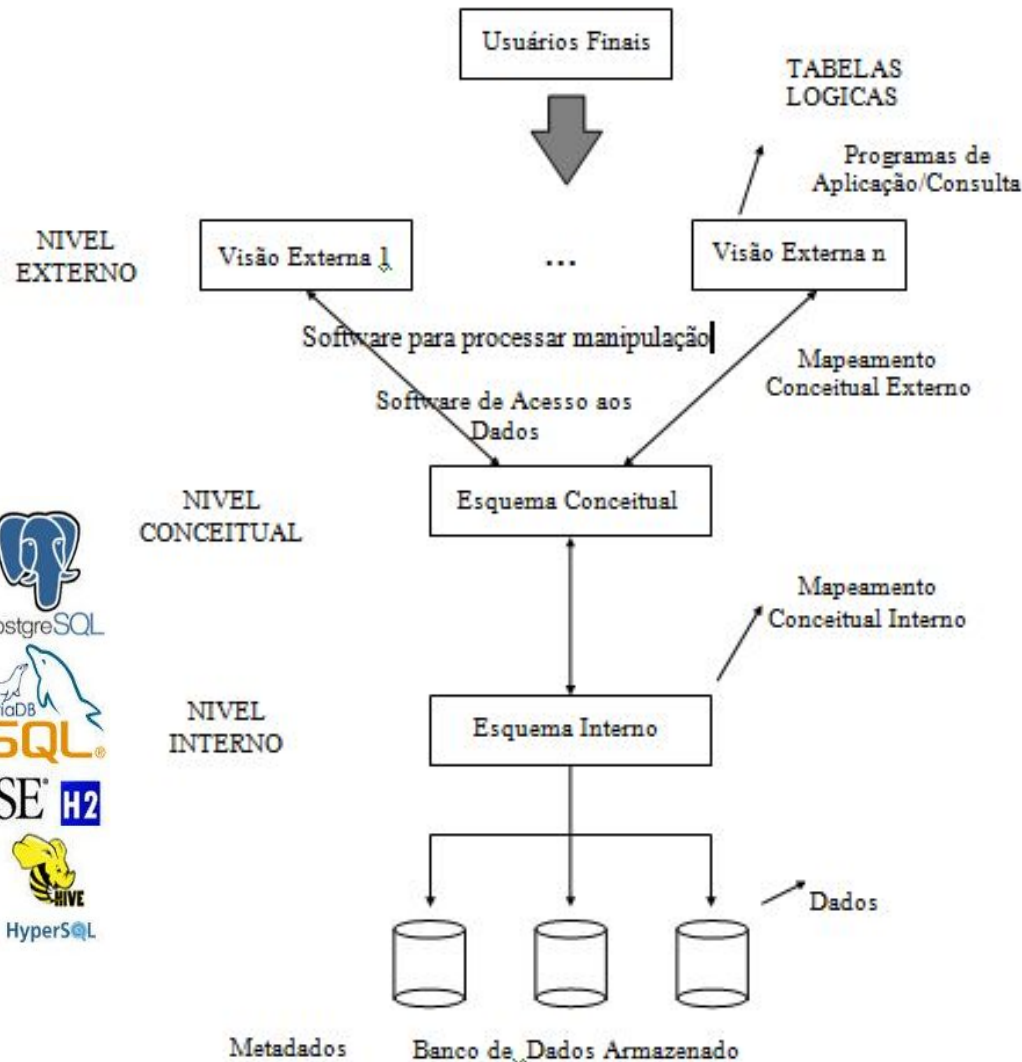
SGBD - SQL - 1. Parte

Tb Funcionário		
Código	Nome	Cd cargo
001	João	01
002	Maria	02
003	Patrícia	03

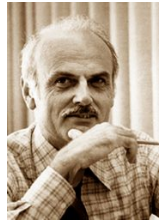
Ta Cargo	
Cd cargo	Cargo
01	Gerente
02	Vendedor
03	Contador



Representação Física do SGBD



SQL - Structured Query Language



- A American National Standards Institute (ANSI) padronizou o SQL em 1986, seguida pela ISO em 1987.
- O SQL passou por revisões em 1992 (SQL-92), 1999 (SQL:1999), 2003 (SQL:2003), e nas versões subsequentes de 2006, 2008, 2011, 2016 e 2019 (SQL:2019).

```
INSERT INTO My_table  
    (field1, field2, field3)  
VALUES  
    ('test', 'N', NULL);
```

```
UPDATE My_table  
    SET field1 = 'updated value'  
    WHERE field2 = 'N';
```

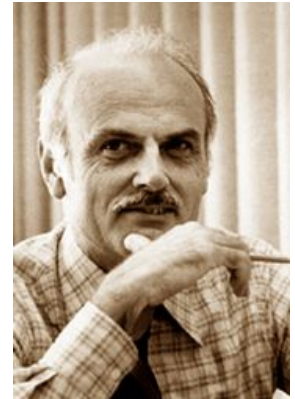
```
DELETE FROM My_table  
    WHERE field2 = 'N';
```

```
select * from Funcionario where Cargo='Analista de Sistemas'
```

Resultados		Mensagens	
Codigo	Nome	Salario	Cargo
1	Djonathan Mastrodi	1300.00	Analista de Sistemas
2	Jose da Silva	1100.00	Analista de Sistemas

SQL - Structured Query Language

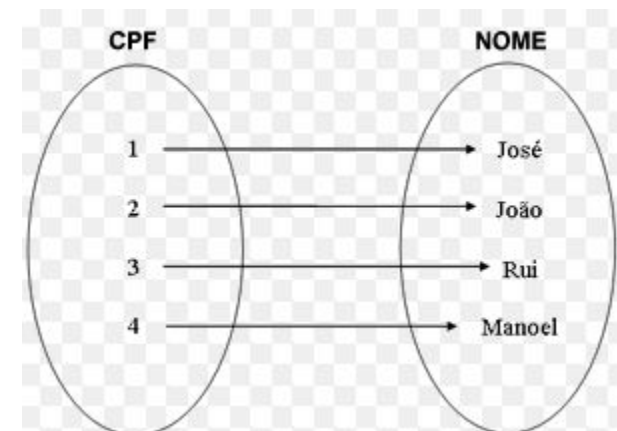
- Surgiu dos trabalhos teóricos de **Edgar Frank Codd** (IBM) procurando um modelo lógico independente de detalhes de implementação.
- Teoria dos conjuntos
- Baseado na álgebra relacional.



Empregado		
Nome	Empld	DeptNome
Harry	3415	Finanças
Sally	2241	Vendas
George	3401	Finanças
Harriet	2202	Vendas

Departamento	
DeptNome	Gerente
Finanças	George
Vendas	Harriet
Produção	Charles

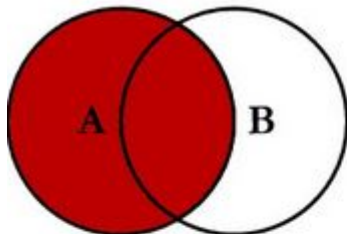
Empregado ⋈ Departamento			
Nome	Empld	DeptNome	Gerente
Harry	3415	Finanças	George
Sally	2241	Vendas	Harriet
George	3401	Finanças	George
Harriet	2202	Vendas	Harriet



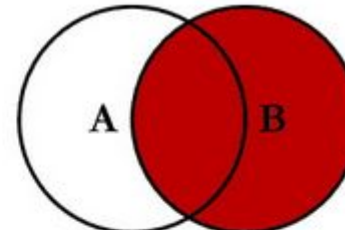
SQL - Structured Query Language

Teoria dos conjuntos

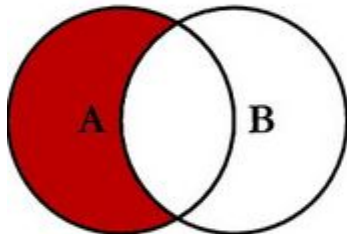
SQL JOINS



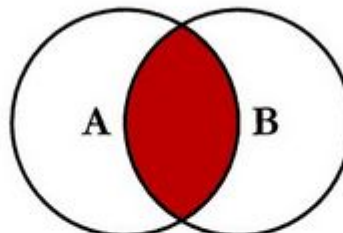
```
SELECT <select_list>  
FROM TableA A  
LEFT JOIN TableB B  
ON A.Key = B.Key
```



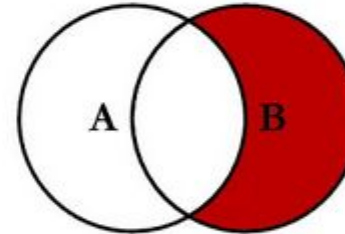
```
SELECT <select_list>  
FROM TableA A  
RIGHT JOIN TableB B  
ON A.Key = B.Key
```



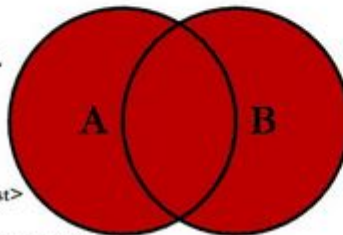
```
SELECT <select_list>  
FROM TableA A  
LEFT JOIN TableB B  
ON A.Key = B.Key  
WHERE B.Key IS NULL
```



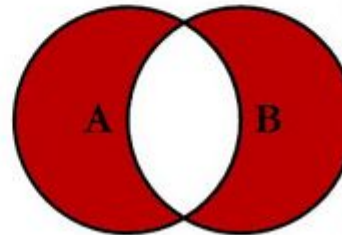
```
SELECT <select_list>  
FROM TableA A  
INNER JOIN TableB B  
ON A.Key = B.Key
```



```
SELECT <select_list>  
FROM TableA A  
RIGHT JOIN TableB B  
ON A.Key = B.Key  
WHERE A.Key IS NULL
```



```
SELECT <select_list>  
FROM TableA A  
FULL OUTER JOIN TableB B  
ON A.Key = B.Key
```



```
SELECT <select_list>  
FROM TableA A  
FULL OUTER JOIN TableB B  
ON A.Key = B.Key  
WHERE A.Key IS NULL  
OR B.Key IS NULL
```

© C.L. Moffatt, 2008

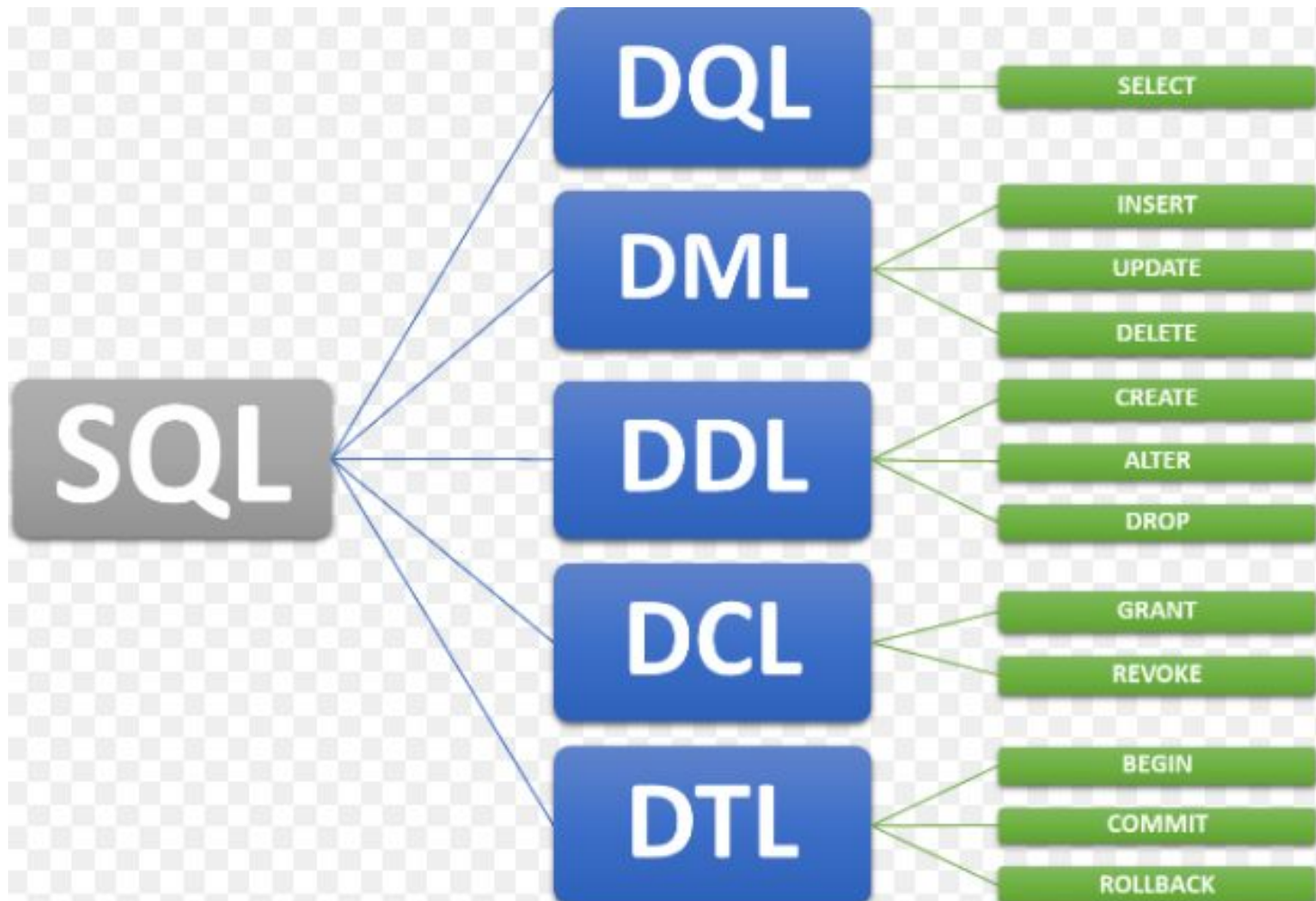
Banco de dados Relacional

Álgebra relacional

<i>Símbolo</i>	<i>Operação</i>	<i>Sintaxe</i>	<i>Tipo</i>
σ	Seleção / Restrição	$\sigma_{\text{condição}} (\text{Relação})$	Primitiva
π	Projeção	$\pi_{\text{expressões}} (\text{Relação})$	Primitiva
\cup	União	$\text{Relação1} \cup \text{Relação2}$	Primitiva
\cap	Intersecção	$\text{Relação1} \cap \text{Relação2}$	Adicional
-	Diferença de conjuntos	$\text{Relação1} - \text{Relação2}$	Primitiva
x	Produto cartesiano	$\text{Relação1} \times \text{Relação2}$	Primitiva
x	Junção	$\text{Relação1} \bowtie \text{Relação2}$	Adicional
\div	Divisão	$\text{Relação1} \div \text{Relação2}$	Adicional
ρ	Renomeação	$\rho_{\text{nome}} (\text{Relação})$	Primitiva
\leftarrow	Atribuição	$\text{variável} \leftarrow \text{Relação}$	Adicional

Categorias dos comandos SQL

Categorias dos comandos SQL



Linguagem de consulta de dados (DQL)

- ✓ DQL é a linguagem de consulta de dados em SQL. O principal comando é o SELECT, usado para recuperar dados de uma ou mais tabelas.

Comando SELECT

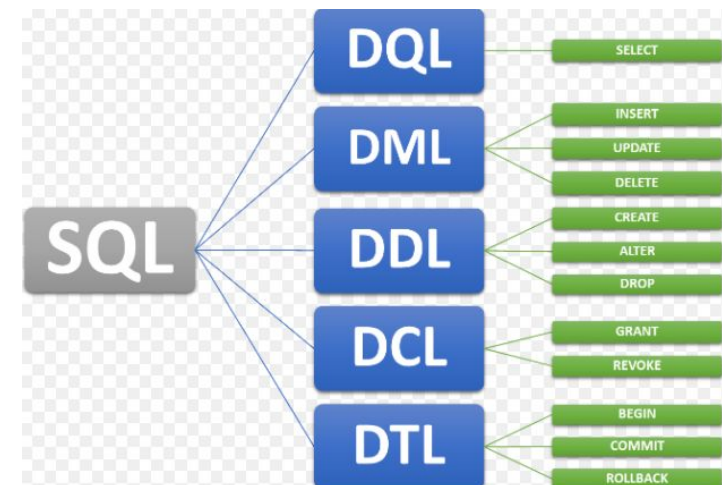
- Selecionando todas as colunas da Tabela

```
SELECT *  
FROM vendedor;
```

- Listar todo o conteúdo de vendedor.

	FirstName
1	Gustavo
2	Catherine
3	Kim
4	Humberto
5	Pilar
6	Frances
7	Margaret
8	Carla
9	Jay
10	Ronald

	FirstName
1	GUSTAVO
2	CATHERINE
3	KIM
4	HUMBERTO
5	PILAR
6	FRANCES
7	MARGARET
8	CARLA
9	JAY
10	RONALD

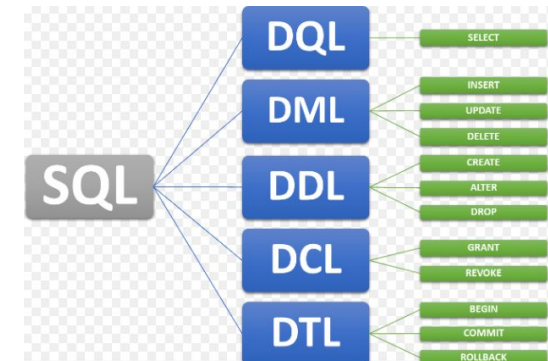


Linguagem de manipulação dos dados (**DML**)

- ✓ O DML manipula dados nas tabelas com três comandos principais: INSERT (inserir), UPDATE (atualizar) e DELETE (deletar).

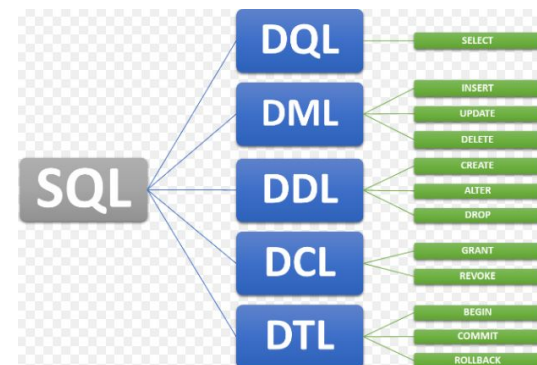
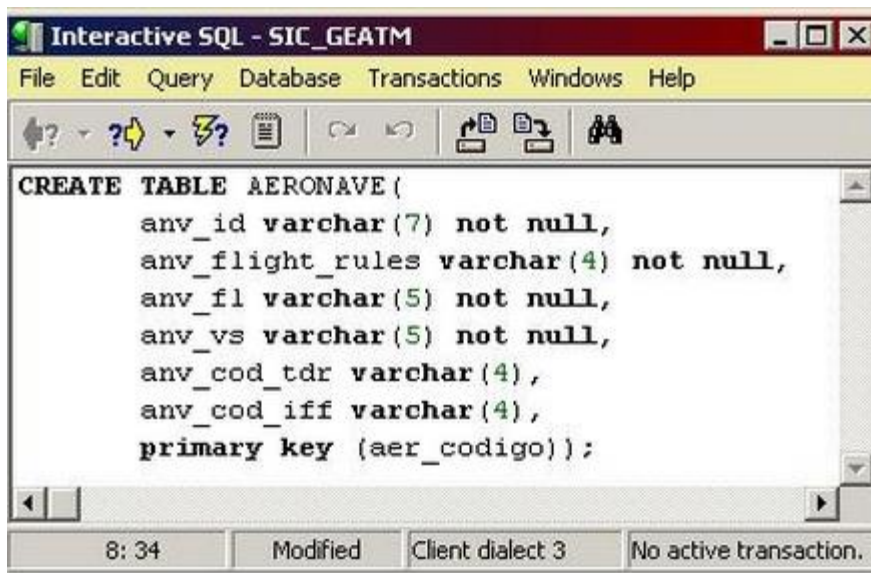
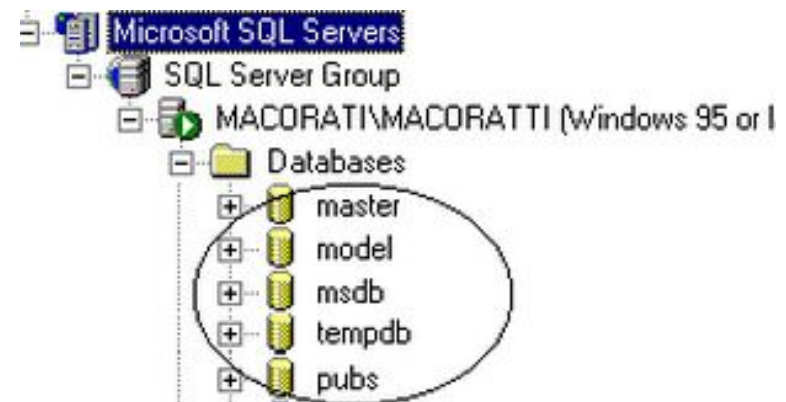


	DepartmentID	DepartmentName	GroupName	CategoryName
1	1	Cardiology	Heart Center	Clinical
2	2	Central ICU	Emergency	Clinical
3	3	Trauma and Emergency Surgery	Emergency	Clinical
4	4	Communications	Administration	Operational
5	5	Oncology	Internal Medicine	Clinical
6	6	Neurology	Internal Medicine	Clinical



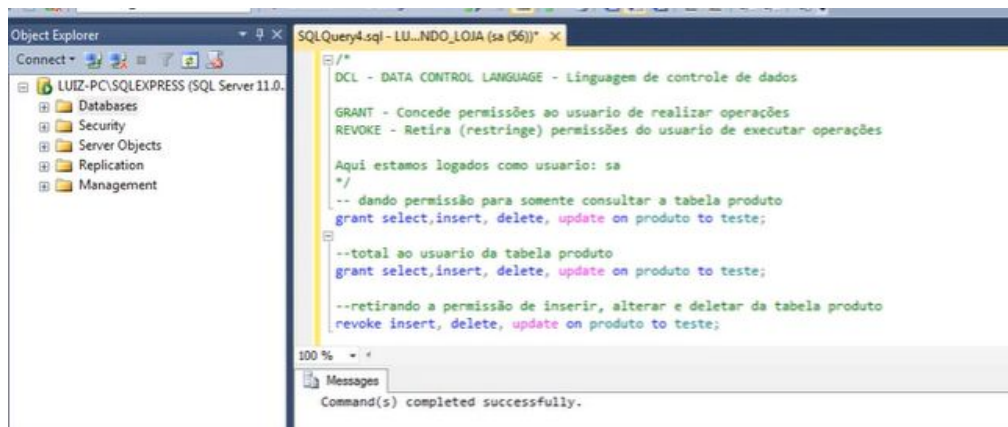
Linguagem de definição dos dados (DDL)

- ✓ A DDL define a estrutura do banco de dados, criando e atualizando arquivos e campos. É usada para definir aplicações, arquivos e campos do banco.



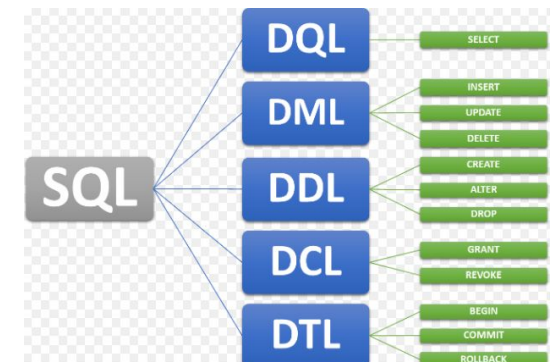
Linguagem de controle dos dados (**DCL**)

- Usado para **controle de acesso** aos dados em um banco de dados.
- Todos acessos podem ser sobre, por exemplo, tabelas, views, functions.



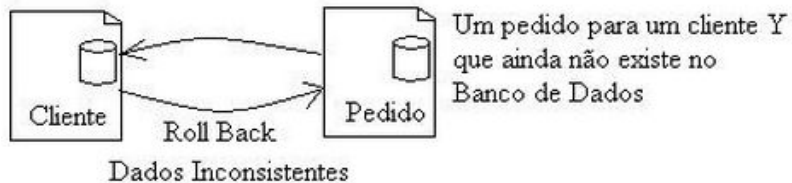
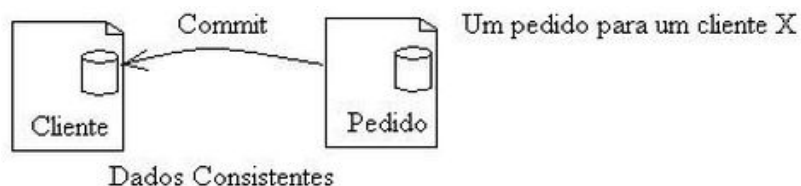
```
SQLQuery4.sql - LU...NDO_LOJA (sa (56)) *  
/*  
DCL - DATA CONTROL LANGUAGE - Linguagem de controle de dados  
GRANT - Concede permissões ao usuário de realizar operações  
REVOKE - Retira (restringe) permissões do usuário de executar operações  
  
Aqui estamos logados como usuário: sa  
*/  
-- dando permissão para somente consultar a tabela produto  
grant select,insert, delete, update on produto to teste;  
  
--total ao usuario da tabela produto  
grant select,insert, delete, update on produto to teste;  
  
--retirando a permissão de inserir, alterar e deletar da tabela produto  
revoke insert, delete, update on produto to teste;
```

100 %
Messages
Command(s) completed successfully.



Linguagem de transação de dados (DTL)

- A **DTL** ou (**Data Transaction Language**) é um subconjunto do SQL para transação de dados. A DTL envolve gerenciamento e controle de transações.



Exercício - Prático

Exercício 01 – Fazer o MER e criar o banco de dados

■ Recursos Humanos

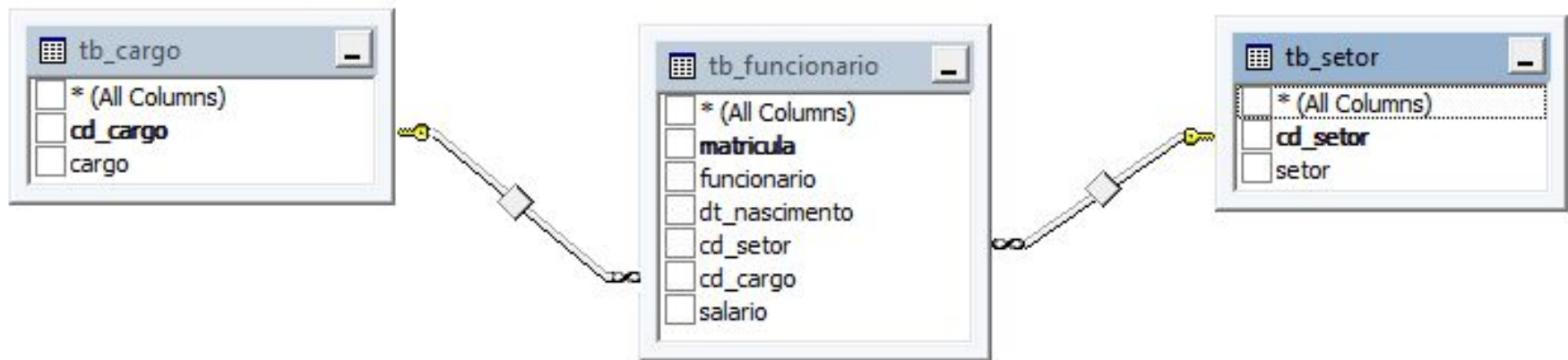
- A empresa de pesquisa agropecuária de referência no Brasil precisa controlar seus colaboradores, as informações básicas.

■ Dados levantados:

- Funcionário
- Data de nascimento
- Cargo – 1 funcionário só pode ter 1 cargo
- setor – 1 funcionário só pode trabalho em 1 setor
- Salário

Exercício – Criar o Banco de Dados no SQL Server

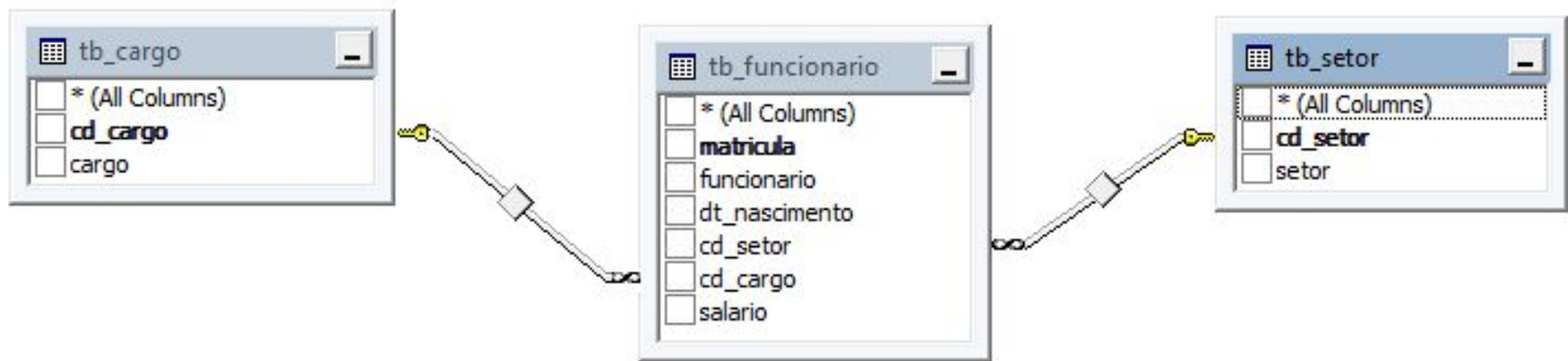
Criar o Banco de Dados



Comandos em SQL - DDL

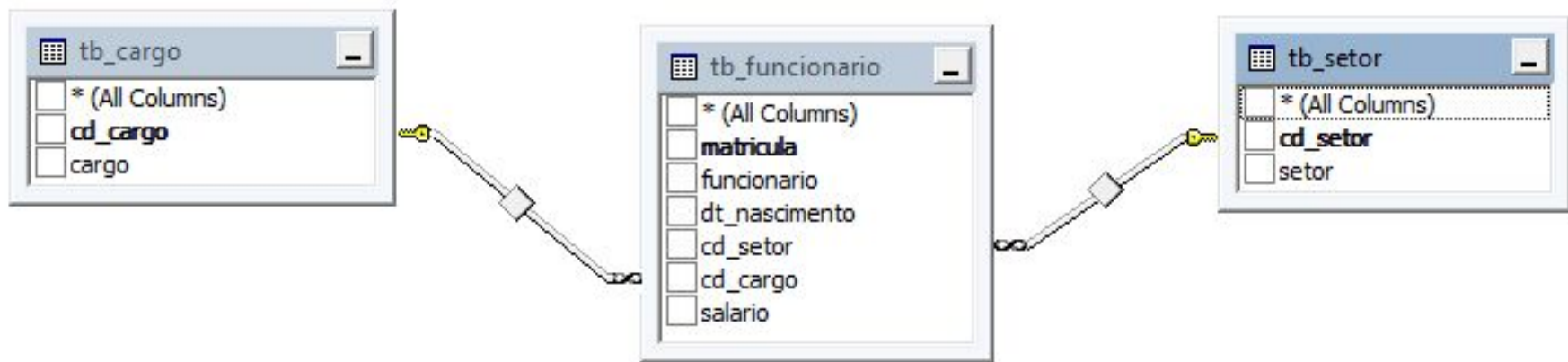
1. Criar as tabelas
2. Tipo de dados dos campos possíveis
3. Campo Chave Primária
4. Criar o relacionamento

Criar o Banco de Dados



- Create database bd_rh;

Usar o Banco de Dados



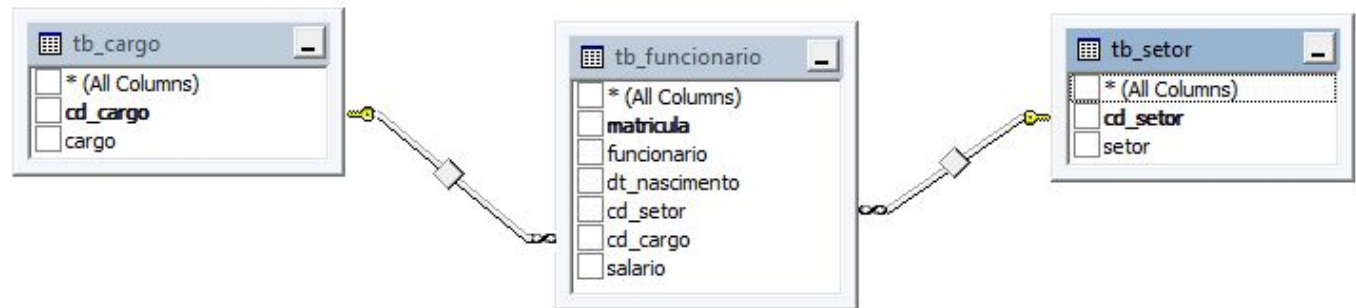
- use bd_rh;

Criar a Tabela tb_funcionario



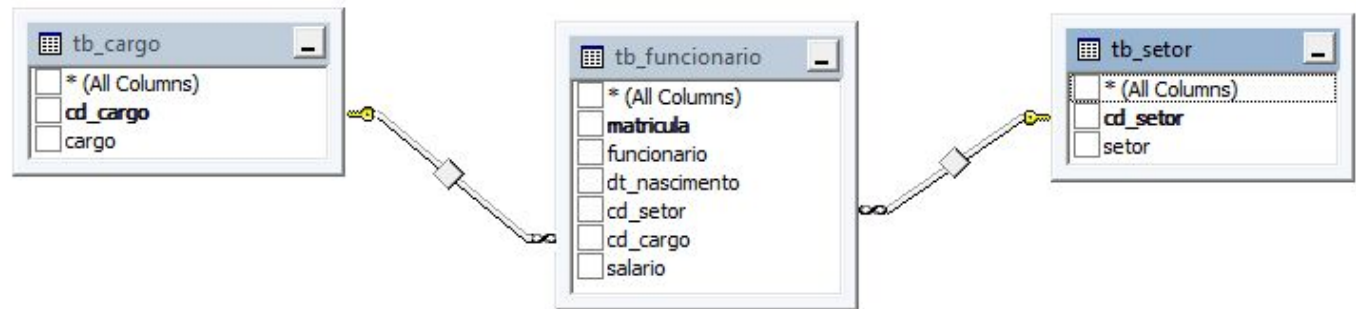
```
create table tb_funcionario
(matricula int not null primary key,
funcionario char(50),
dt_nascimento date,
cd_setor int,
cd_cargo int,
salario decimal(8,2));
```

Criar a Tabela tb_cargo



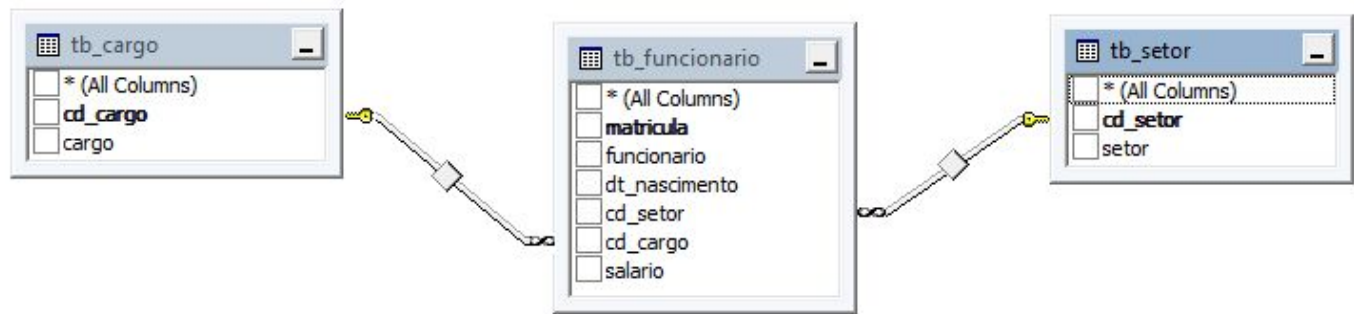
```
create table tb_cargo
(cd_cargo int not null primary key,
cargo char(20));
```

Criar a Tabela tb_setor



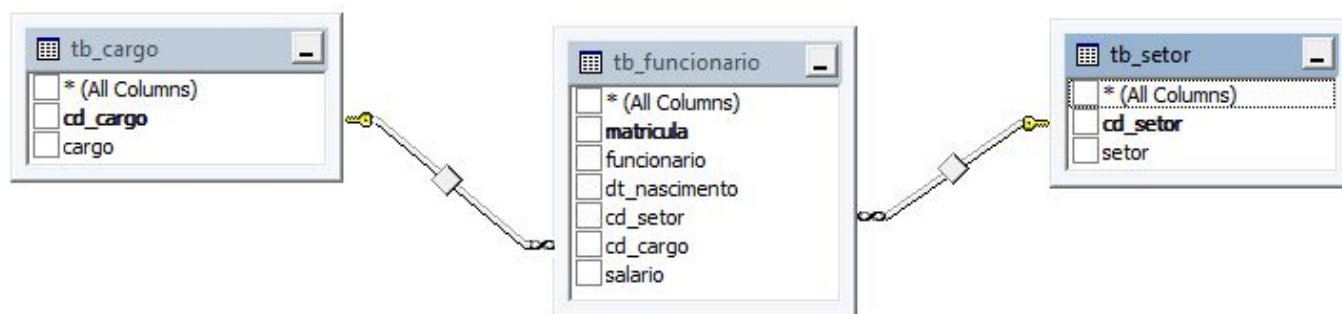
```
create table tb_setor
(cd_setor int not null primary key,
setor char(30));
```

Criar o relacionamento tb_funcionario x tb_setor



```
alter table tb_funcionario  
add constraint fk_setor foreign key (cd_setor)  
references tb_setor (cd_setor);
```

Criar o relacionamento tb_funcionario x tb_cargo



```
alter table tb_funcionario  
add constraint fk_cargo foreign key (cd_cargo)  
references tb_cargo (cd_cargo);
```

Álgebra Relacional

Operadores

- Podem ser agrupadas em duas categorias:
 - Operadores Relacionais:
 - Seleção
 - Projeção
 - Junção
 - Divisão
 - Operadores Tradicionais:
 - União
 - Interseção
 - Diferença
 - Produto Cartesiano

Operadores Relacionais

Operadores Relacionais

- **SELECT:**
 - Símbolo σ ([sigma](#)).
 - Extrai tuplas específicas de uma relação específica.
 - **Select * from tb_funcionario;**

```
select * from tb_funcionario;
```

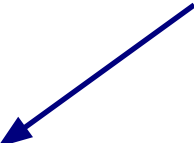
100 %

Resultados Mensagens

	matricula	funcionario	cd_sexo	dt_nascimento	cd_estado_civil	cd_setor	cd_cargo	salario	estado_civil
1	1	Ana Clara	F	1977-05-07 00:00:00.000	1	5	1	3000,00	Solteiro
2	2	Patricia Azevedo	F	1944-04-07 00:00:00.000	1	1	1	4000,00	Solteiro
3	3	Jose Maria	M	1971-10-05 00:00:00.000	1	3	1	6000,00	Solteiro
4	7	Jose Pereira	M	1980-10-09 00:00:00.000	2	3	1	500,00	Casado(a)
5	10	Paula Nobrega	F	1977-12-03 00:00:00.000	3	4	2	3000,00	Separado

Operadores Relacionais

- **SELECT:**



```
select * from tb_funcionario where matricula = 1;
```

100 %

Resultados Mensagens

	matricula	funcionario	cd_sexo	dt_nascimento	cd_estado_civil	cd_setor	cd_cargo	salario	estado_civil
1	1	Ana Clara	F	1977-05-07 00:00:00.000	1	5	1	3000,00	Solteiro

Operadores Relacionais

■ Project:

- Símbolo Π (pi).
- Extrai determinadas colunas de uma tabela e descarta outras.
- **select matricula, funcionario, dt_nascimento from tb_funcionario**

```
select matricula, funcionario, dt_nascimento from tb_funcionario;
```

100 % <

Resultados Mensagens

	matricula	funcionario	dt_nascimento
1	1	Ana Clara	1977-05-07 00:00:00.000
2	2	Patricia Azevedo	1944-04-07 00:00:00.000
3	3	Jose Maria	1971-10-05 00:00:00.000
4	7	Jose Pereira	1980-10-09 00:00:00.000
5	10	Paula Nobrega	1977-12-03 00:00:00.000

Operadores Relacionais

■ Project:

```
select * from tb_funcionario where matricula = 1;
```

100 %

Resultados Mensagens

	matricula	funcionario	cd_sexo	dt_nascimento	cd_estado_civil	cd_setor	cd_cargo	salario	estado_civil
1	1	Ana Clara	F	1977-05-07 00:00:00.000	1	5	1	3000,00	Solteiro



```
select matricula, funcionario, dt_nascimento from tb_funcionario;
```

100 %

Resultados Mensagens

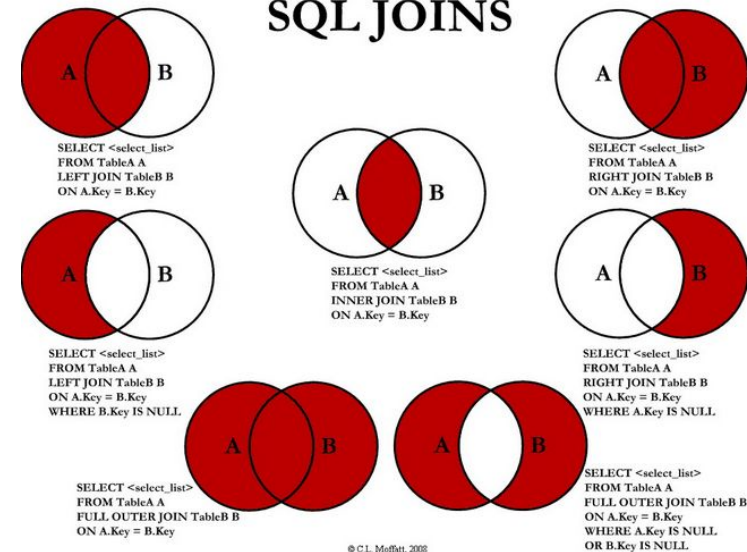
	matricula	funcionario	dt_nascimento
1	1	Ana Clara	1977-05-07 00:00:00.000
2	2	Patricia Azevedo	1944-04-07 00:00:00.000
3	3	Jose Maria	1971-10-05 00:00:00.000
4	7	Jose Pereira	1980-10-09 00:00:00.000
5	10	Paula Nobrega	1977-12-03 00:00:00.000

Operadores Relacionais

■ JOIN (INNER JOIN):

- Símbolo [X].
- Constrói uma relação a partir de duas relações específicas consistindo em todas as possibilidades de pares de tuplas concatenados, uma de cada uma das duas relações específicas, de forma que em cada par as duas tuplas satisfaçam uma condição específica.

SQL JOINS



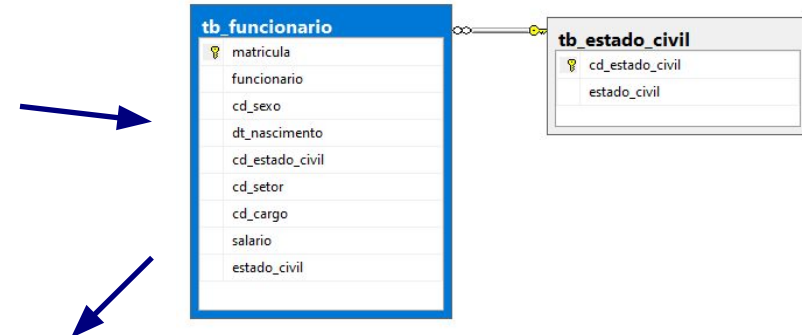
Operadores Relacionais

```
select matricula, funcionario, cd_estado_civil from tb_funcionario;
```

100 %

Resultados Mensagens

	matricula	funcionario	cd_estado_civil
1	1	Ana Clara	1
2	2	Patricia Azevedo	1
3	3	Jose Maria	1
4	7	Jose Pereira	2
5	10	Paula Nobrega	3



```
select F.matricula, F.funcionario, EC.estado_civil from tb_funcionario F
inner join tb_estado_civil EC
on F.cd_estado_civil = EC.cd_estado_civil;
```

100 %

Resultados Mensagens

	matricula	funcionario	estado_civil
1	1	Ana Clara	Solteiro(a)
2	2	Patricia Azevedo	Solteiro(a)
3	3	Jose Maria	Solteiro(a)
4	7	Jose Pereira	Casado(a)
5	10	Paula Nobrega	Separado(a)

DML – Linguagem de Manipulação de dados

Insert, update, delete e select

Operadores

Insert,

select,

update,

delete



CREATE

READ

UPDATE

DELETE

C

R

U

D

INSERT

```

insert into tb_funcionario
(matricula, funcionario, cd_sexo, dt_nascimento, cd_estado_civil, cd_setor, cd_cargo, salario)
values
(4, 'Carlos', 'M', '1990-08-29', 1, 2, 3, 10000);
  
```

100 %

Resultados Mensagens

	matricula	funcionario	cd_sexo	dt_nascimento	cd_estado_civil	cd_setor	cd_cargo	salario
1	1	Ana Clara	F	1977-05-07 00:00:00.000	1	5	1	3000,00
2	2	Patricia Azevedo	F	1944-04-07 00:00:00.000	1	1	1	4000,00
3	3	Jose Maria	M	1971-10-05 00:00:00.000	1	3	1	6000,00
4	7	Jose Pereira	M	1980-10-09 00:00:00.000	2	3	1	500,00
5	10	Paula Nobrega	F	1977-12-03 00:00:00.000	3	4	2	3000,00

```

select * from tb_funcionario;
  
```

100 %

Resultados Mensagens

	matricula	funcionario	cd_sexo	dt_nascimento	cd_estado_civil	cd_setor	cd_cargo	salario
1	1	Ana Clara	F	1977-05-07 00:00:00.000	1	5	1	3000,00
2	2	Patricia Azevedo	F	1944-04-07 00:00:00.000	1	1	1	4000,00
3	3	Jose Maria	M	1971-10-05 00:00:00.000	1	3	1	6000,00
4	4	Carlos	M	1990-08-29 00:00:00.000	1	2	3	10000,00
5	7	Jose Pereira	M	1980-10-09 00:00:00.000	2	3	1	500,00
6	10	Paula Nobrega	F	1977-12-03 00:00:00.000	3	4	2	3000,00

INSERT

```
insert into tb_estado_civil  
(cd_estado_civil, estado_civil)  
values  
(1, 'Solteiro(a)'),  
(2, 'Casado(a))')
```

```
select * from tb_estado_civil;
```

100 %



Resultados



Mensagens

	cd_estado_civil	estado_civil
1	1	Solteiro(a)
2	2	Casado(a)

INSERT

```
select matricula, funcionario, dt_nascimento, salario from tb_funcionario;
select * from tb_funcionario;
```

100 %



Resultados



Mensagens

	matricula	funcionario	dt_nascimento	salario
1	1	Ana Clara	1977-05-07 00:00:00.000	3000,00
2	2	Patricia Azevedo	1944-04-07 00:00:00.000	4000,00
3	3	Jose Maria	1971-10-05 00:00:00.000	6000,00
4	4	Carlos	1990-08-29 00:00:00.000	10000,00
5	7	Jose Pereira	1980-10-09 00:00:00.000	500,00
6	10	Paula Nobrega	1977-12-03 00:00:00.000	3000,00

UPDATE

```
select * from tb_cargo;
```

100 % <

Resultados Mensagens

	cd_cargo	cargo
1	1	Enfermeiro(a)
2	2	Administrador(a)
3	3	Analista
4	4	Engenheiro(a)
5	5	Advogado(a)
6	6	Gerente
7	7	Executivo(a)

```
update tb_cargo set cargo = 'Enfermeiro'
where cd_cargo = 1;

select * from tb_cargo;
```

100 % <

Resultados Mensagens

	cd_cargo	cargo
1	1	Enfermeiro
2	2	Administrador(a)
3	3	Analista
4	4	Engenheiro(a)
5	5	Advogado(a)
6	6	Gerente
7	7	Executivo(a)

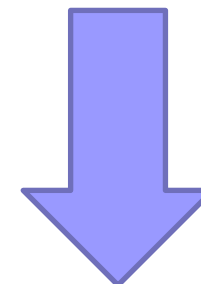
UPDATE

```
select matricula, funcionario, cd_estado_civil, cd_cargo from tb_funcionario;
```

100 %

Resultados Mensagens

	matricula	funcionario	cd_estado_civil	cd_cargo
1	1	Ana Clara	1	1
2	2	Patricia Azevedo	1	1
3	3	Jose Maria	1	1
4	4	Carlos	1	3
5	7	Jose Pereira	2	1
6	10	Paula Nobrega	3	2



```
update tb_funcionario
set cd_estado_civil = 2, cd_cargo = 2
where matricula = 1;

select matricula, funcionario, cd_estado_civil, cd_cargo from tb_funcionario;
```

100 %

Resultados Mensagens

	matricula	funcionario	cd_estado_civil	cd_cargo
1	1	Ana Clara	2	2
2	2	Patricia Azevedo	1	1
3	3	Jose Maria	1	1
4	4	Carlos	1	3
5	7	Jose Pereira	2	1
6	10	Paula Nobrega	3	2

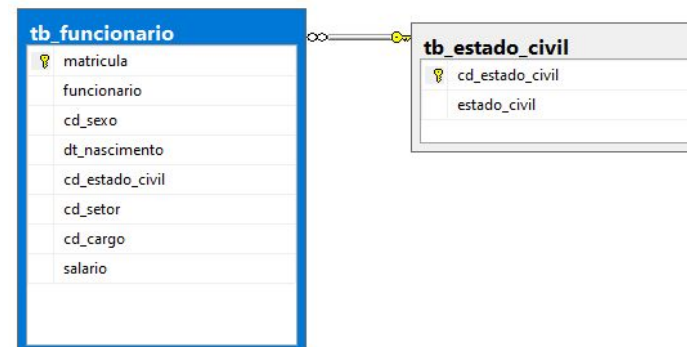
DELETE

```
select matricula, funcionario, cd_estado_civil, cd_cargo from tb_funcionario;
```

100 %

Resultados Mensagens

	matricula	funcionario	cd_estado_civil	cd_cargo
1	1	Ana Clara	1	1
2	2	Patricia Azevedo	1	1
3	3	Jose Maria	1	1
4	4	Carlos	1	3
5	7	Jose Pereira	2	1
6	10	Paula Nobrega	3	2



```
delete tb_estado_civil
where cd_estado_civil = 1;
```

100 %

Mensagens

Mensagem 547, Nível 16, Estado 0, Linha 1

A instrução DELETE conflitou com a restrição do REFERENCE "fk_estado_civil". O conflito ocorreu no banco de dados "bd_rh100", tabela "dbo.tb_estado_civil".

A instrução foi finalizada.

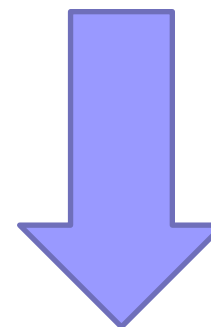
DELETE

```
select matricula, funcionario, cd_estado_civil, cd_cargo from tb_funcionario;
```

100 % <

Resultados Mensagens

	matricula	funcionario	cd_estado_civil	cd_cargo
1	1	Ana Clara	2	2
2	2	Patricia Azevedo	1	1
3	3	Jose Maria	1	1
4	4	Carlos	1	3
5	7	Jose Pereira	2	1
6	10	Paula Nobrega	3	2



```
delete tb_funcionario  
where matricula = 4;
```

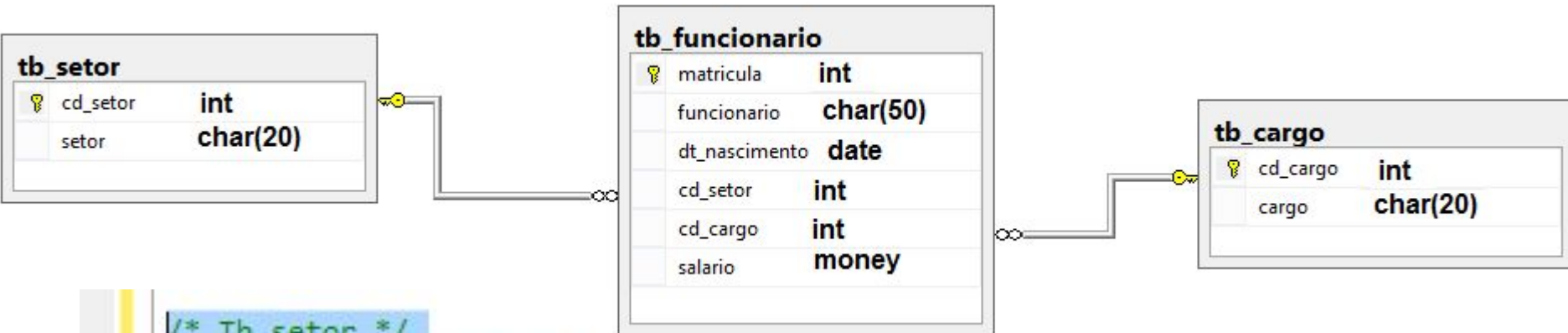
100 % <

Resultados Mensagens

	matricula	funcionario	cd_estado_civil	cd_cargo
1	1	Ana Clara	2	2
2	2	Patricia Azevedo	1	1
3	3	Jose Maria	1	1
4	7	Jose Pereira	2	1
5	10	Paula Nobrega	3	2

Comandos para Popular o Banco de dados BD_RH

Popular o Banco de dados BD_RH



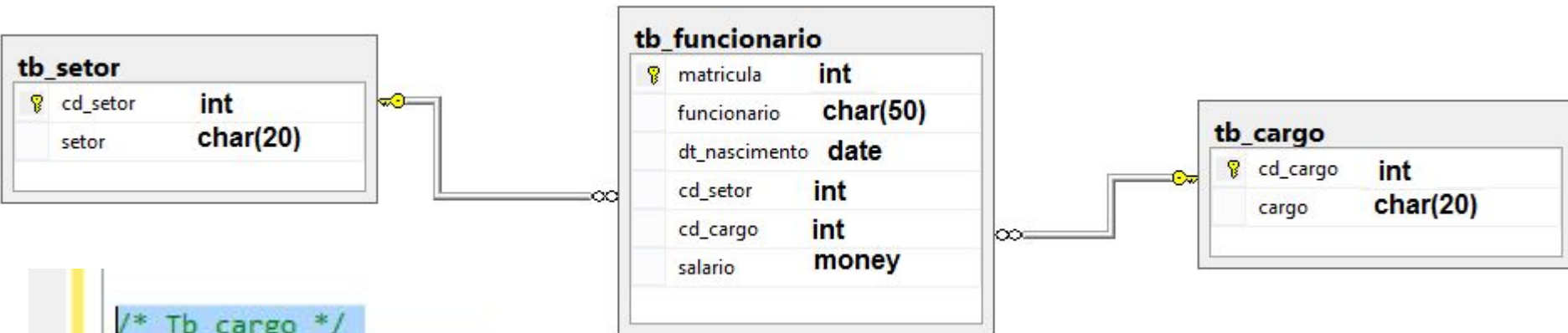
```
/* Tb_setor */  
select * from tb_setor
```

100 %

Resultados Mensagens

	cd_setor	setor
1	1	Enfermaria
2	2	Administracao
3	3	Informatica
4	4	Engenharia
5	5	Juridico
6	6	Logistica
7	7	Presidencia

Popular o Banco de dados BD_RH



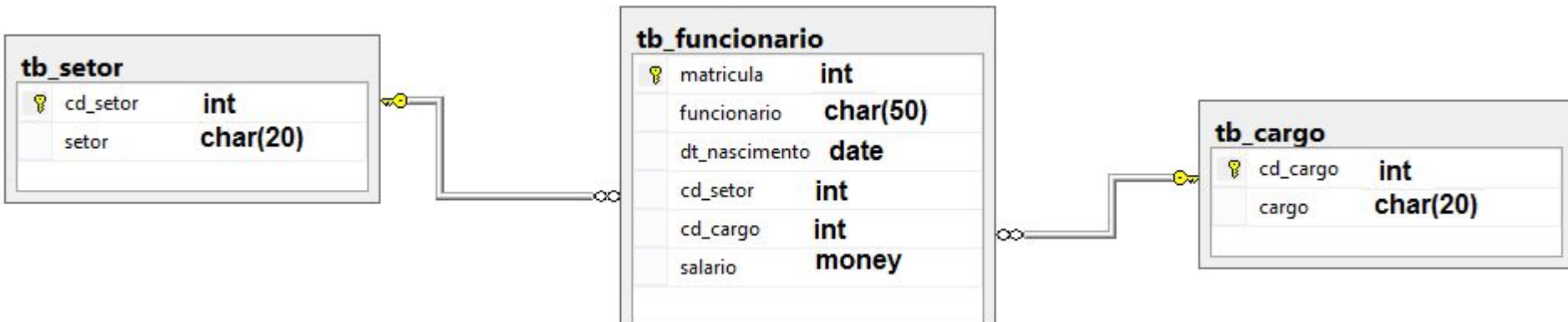
```
/* Tb_cargo */  
select * from tb_cargo
```

100 %

Resultados Mensagens

	cd_cargo	cargo
1	1	Enfermeiro(a)
2	2	Administrador(a)
3	3	Analista
4	4	Engenheiro(a)
5	5	Advogado(a)
6	6	Gerente
7	7	Executivo(a)

Popular o Banco de dados BD_RH



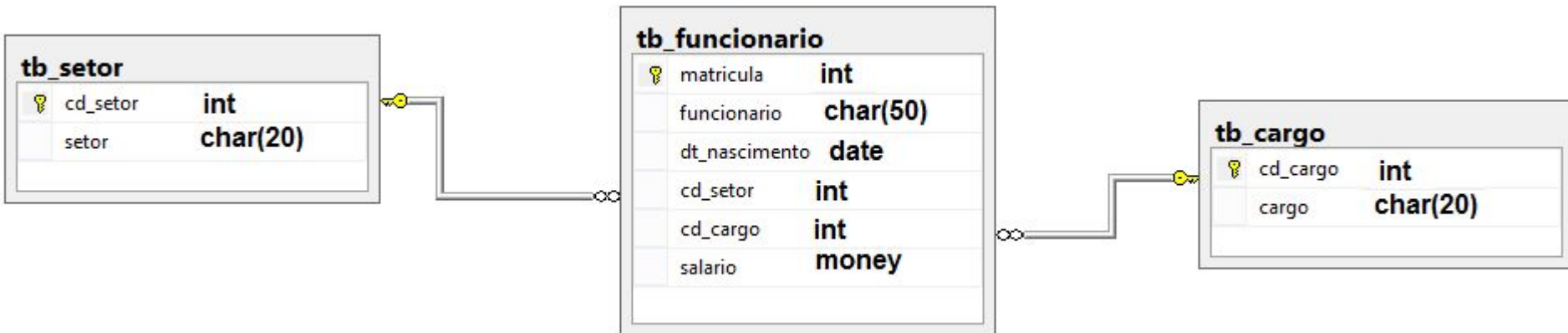
```
/* Tb_Funcionario */
select * from tb_funcionario
```

100 %

Resultados Mensagens

	matricula	funcionario	dt_nascimento	cd_setor	cd_cargo	salario
1	1	Ana Clara	1977-07-05	5	1	3000,00
2	2	Patricia Azevedo	1944-07-04	1	1	4000,00
3	3	Jose Maria	1971-05-10	3	1	6000,00
4	4	Sonia Abrantes	1979-05-29	4	1	7000,00
5	5	Valdir Reinaldo	1960-09-22	2	2	16000,00
6	6	Jose Alberto	1955-01-13	2	2	15000,00

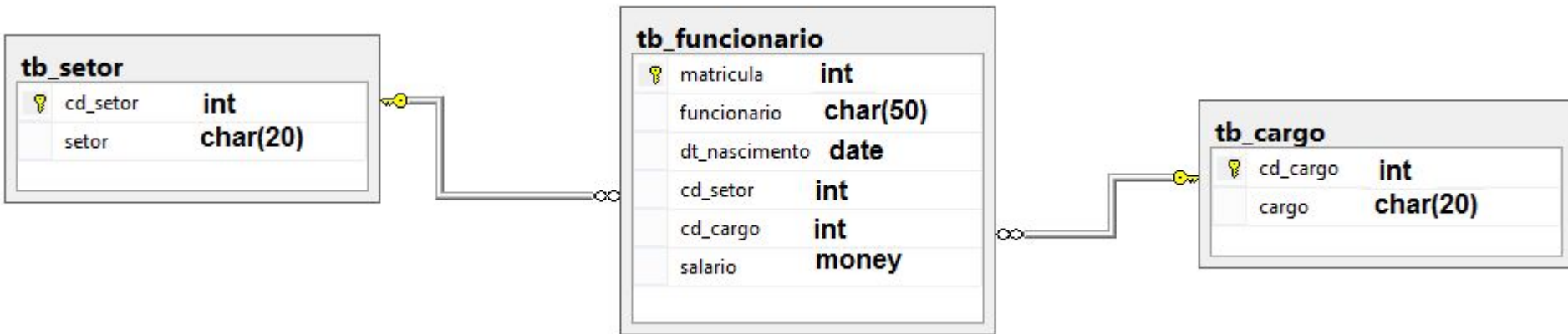
Popular o Banco de dados BD_RH



```
insert into tb_setor (cd_setor, setor)
values
```

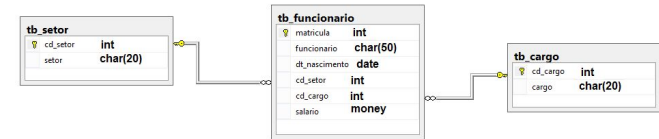
```
(1, 'Enfermaria'),
(2, 'Administracao'),
(3, 'Informatica'),
(4, 'Engenharia'),
(5, 'Juridico'),
(6, 'Logistica'),
(7, 'Presidencia');
```

Popular o Banco de dados BD_RH



```
insert into tb_cargo
(cd_cargo, cargo)
values
(1, 'Enfermeiro(a)'),
(2, 'Administrador(a)' ) ,
(3, 'Analista' ) ,
(4, 'Engenheiro(a)' ) ,
(5, 'Advogado(a)' ) ,
(6, 'Gerente' ) ,
(7, 'Executivo(a)' ) ;
```

Popular o Banco de dados BD_RH



```
insert into tb_funcionario
```

```
(matricula, funcionario, dt_nascimento, cd_setor, cd_cargo, salario)
```

```
Values
```

```
(1, 'Ana Clara', '1977-07-05', 5, 1, 3000),
(2, 'Patricia Azevedo', '1944-07-04', 1, 1, 4000),
(3, 'Jose Maria', '1971-05-10', 3, 1, 6000),
(4, 'Sonia Abrantes', '1979-05-29', 4, 1, 7000),
(5, 'Valdir Reinaldo', '1960-09-22', 2, 2, 16000),
(6, 'Jose Alberto', '1955-01-13', 2, 2, 15000);
```

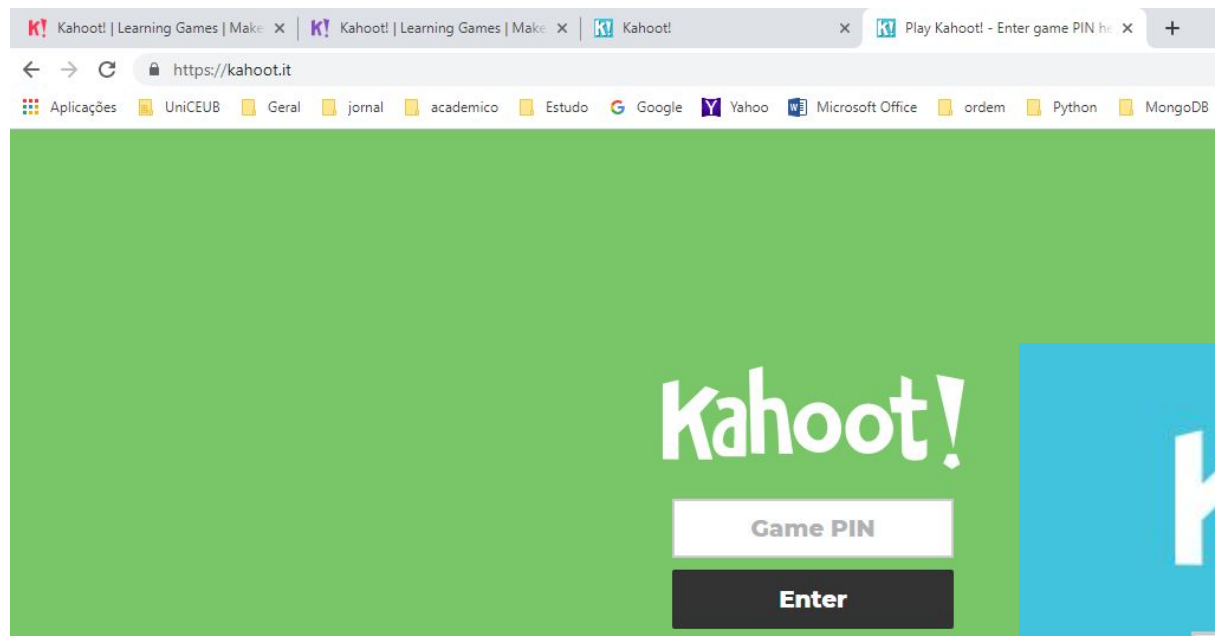
Exercício

exercício

- 1) Excluir cd_cargo = 1 na tabela tb_cargo**
- 2) Alterar – aumentar os salarios dos tb_funcionarios em 10%**
- 3) Listar a tabela (tb_funcionario) salario > 5000**

Questionário

Questionário



<https://kahoot.it/>

Nome + RA



Agenda

- **SGBD - Sistema Gerenciador de Banco de Dados**
- **Representação Física do SGBD**
- **SQL - Structured Query Language**
- **Categorias do SQL**
- **Exercício prático**

Dúvidas

Jose.wellington@uniceub.br