

# ***Bases de Dados***

**Pedro Furtado**

Departamento de Engenharia Informática  
Faculdade de Ciências e Tecnologia da Universidade de Coimbra  
2018/2019

Introdução

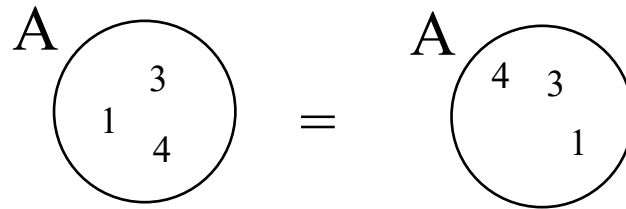
# SET THEORY AND RELATIONAL ALGEBRA

# How SLQ relates to maths...

Select emp.nome, sal,funcao,dep.nome  
From emp join dep on (emp.ndep=dep.ndep)  
Where emp.sal>200 and dep.ndep=10;

$\pi_{\text{emp.nome, sal,funcao,dep.nome}} [\sigma_{\text{emp.sal} > 200} (\sigma_{\text{dep.ndep} = 10} ( \text{emp} \bowtie_{\text{ndep=ndep}} \text{dep} ) ) ]$

# Os conjuntos não têm ordem definida



$$\{2,3,4\} = \{4,3,2\}$$

# The concept of relation=table ...

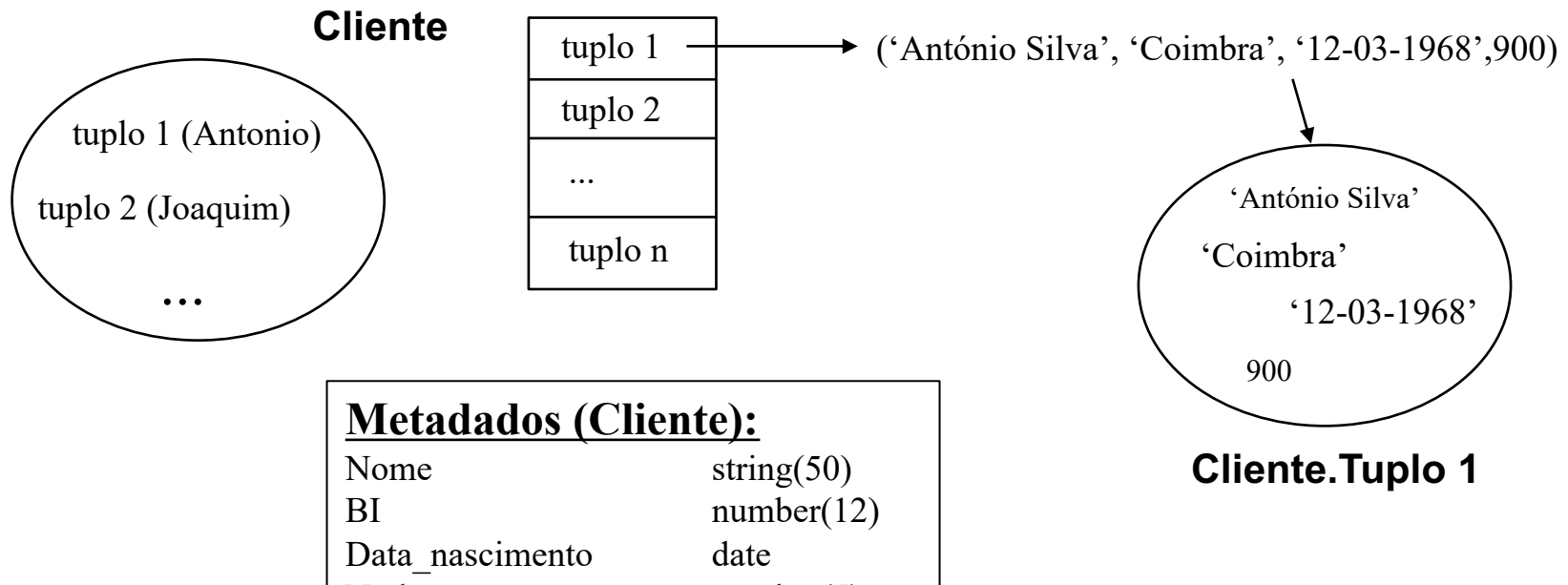
- It is a **set of rows**
- **Sets are unordered, sequences are ordered**

# The relation=table is a set of tuples...

## A tuple is a set of attributes

**Cliente**

| Nome               | Cidade  | BI      | Data_nascimento | N_de_conta |
|--------------------|---------|---------|-----------------|------------|
| António Silva      | Coimbra | 1234343 | 12-03-1968      | 900        |
| Joaquim Alves Dias | Guarda  | 1256673 | 18-01-1978      | 556        |
| ...                |         |         |                 |            |



- Why is it there no order in tables? Wouldn't it be better to have an order?
- And if it's big, how do I get speed?
- So how do we guarantee speed of access to specific data?

- SQL is an interface language for humans to write set operations



# Set Theory (relational algebra)

- Set = collection of objects (members or elements).
- All mathematical objects can be put into sets.

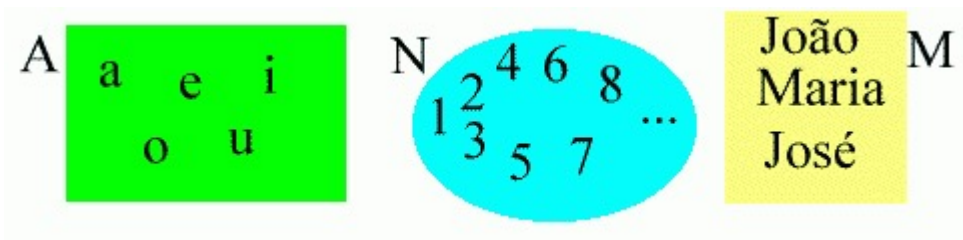
# Notação

- Pertença (belongs):  $1 \in \mathbb{N}$ ,  $0 \notin \mathbb{N}$
- Diagrama de Venn-Euler: (lê-se: "Ven-óiler") Os conjuntos são mostrados graficamente.

$A = \{a, e, i, o, u\}$

$N = \{1, 2, 3, 4, \dots\}$

$M = \{\text{João}, \text{Maria}, \text{José}\}$

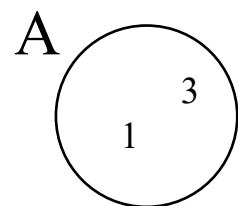


# Set Theory (relational algebra)

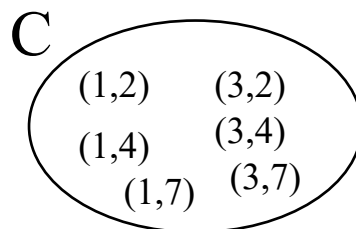
Set = collection of objects (members or elements).

Given **object O** (the object can be a set it itself) and an **S set**, the object is a set element if the following child relationship occurs:

$$O \in S$$

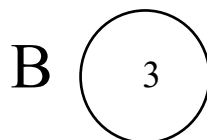
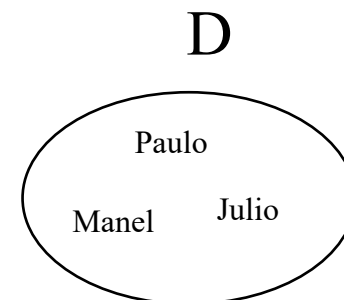


1 e 3 são elementos de A



(1,2) é elemento de C

1 e 2 são elementos de (1,2)



Given sets B and A, the inclusion relationship is  $B \subseteq A$

# Alguns conjuntos especiais

- **Conjunto vazio:**

É um conjunto que não possui elementos  
 $\{ \}$  ou por  $\emptyset$ .

**Exemplo com intersecção:**

Se  $A=\{a,e,i,o,u\}$  e  $B=\{1,2,3,4\}$  então  **$A \cap B = \emptyset$**

- **Conjunto universo:**

É um conjunto que contém todos os elementos =  $U$

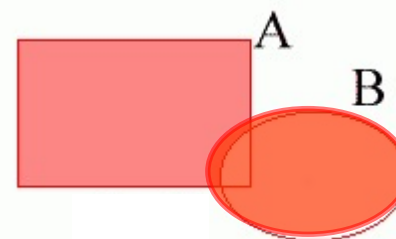
Some operations on sets...

# Operações sobre conjuntos

- **Reunião de conjuntos (UNION)**

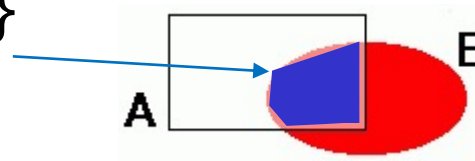
A reunião dos conjuntos A e B é o conjunto de todos os elementos que pertencem ao conjunto A ou ao conjunto B.

$$A \cup B = \{ x: x \in A \vee x \in B \}$$



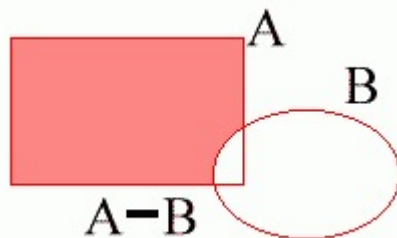
- **Interseção de conjuntos (intersection)**

$$A \cap B = \{ x: x \in A \wedge x \in B \}$$



- **Diferença de conjuntos (difference)**

A diferença entre os conjuntos A e B é o conjunto de todos os elementos que pertencem ao conjunto A e *não* pertencem ao conjunto B.



# Operations on sets

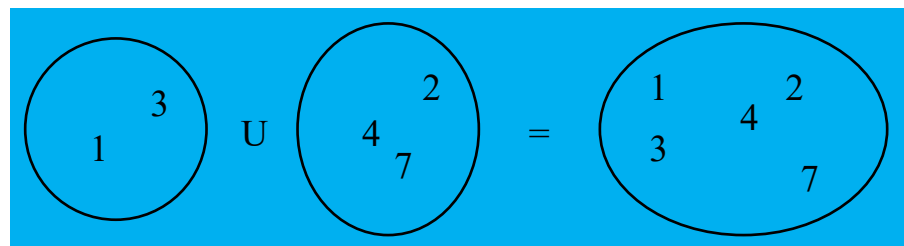
Set = collection of objects (members or elements).

Binary operations on sets:

Intersection, Union,

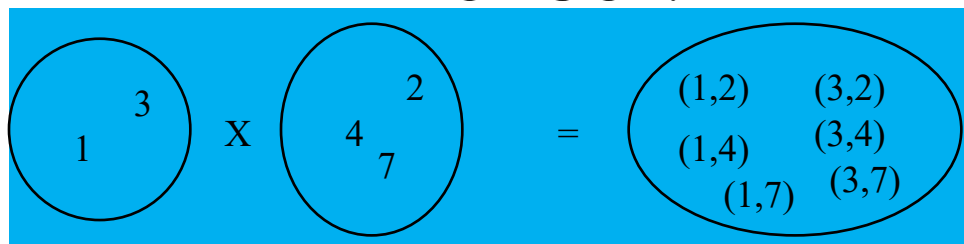
Difference of sets, Cartesian product.

UNION:



The Cartesian product is a set whose members are all possible ordered pairs of members of the two sets participating in the operation.

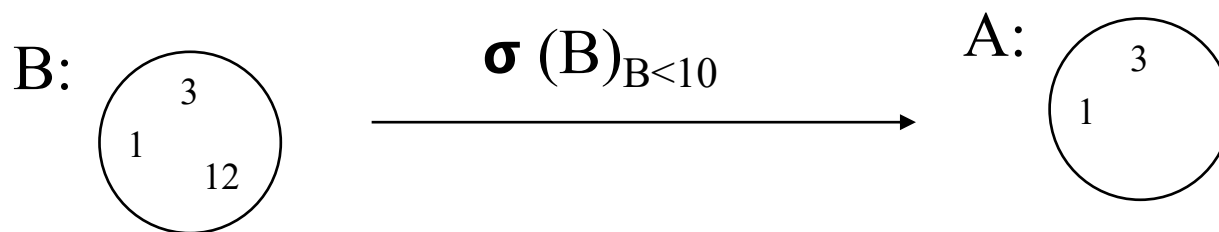
PRODUCT:



# Selection

SELECTION= operation that selects sub-set of B based on a condition

$$A = \sigma(B)_{B < 10}$$



**$\sigma$  = SELECTION**

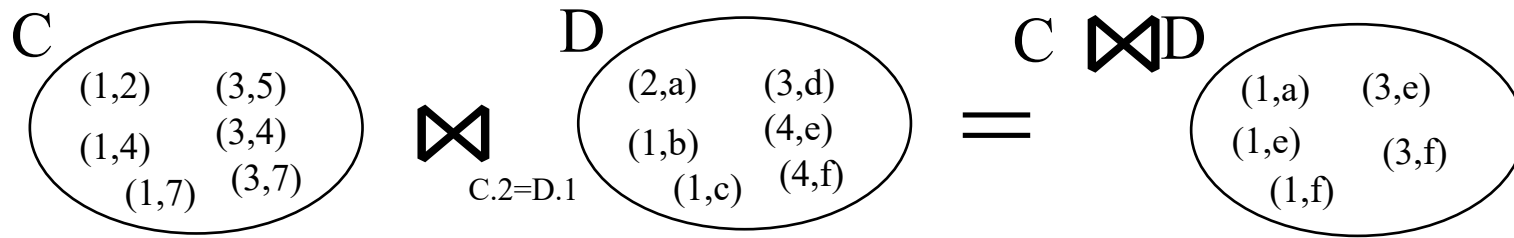


# Projecção

$$\pi_{C.2} \left( C \begin{array}{cc} (1,2) & (3,5) \\ (1,4) & (3,4) \\ (1,7) & (3,7) \end{array} \right) = C \begin{array}{cc} (2) & (5) \\ (4) & (4) \\ (7) & (7) \end{array}$$

**$\pi$  = PROJECTION**

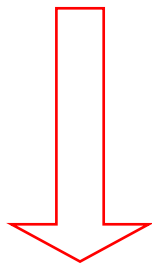
# Join



And some operations on relations...

# Operações mais básicas: **SELECÇÃO** $\sigma$

| Nome               | Cidade  | BI      | Data_nascimento | N_de_conta |
|--------------------|---------|---------|-----------------|------------|
| António Silva      | Coimbra | 1234343 | 12-03-1968      | 900        |
| Joaquim Alves Dias | Guarda  | 1256673 | 18-01-1978      | 556        |
| Maria Teresa Horta | Lisboa  | 1275432 | 15-09-1978      | 900        |
| Joana Antunes      | Lisboa  | 1022634 | 23-07-1988      | 801        |
| Luísa Saraiva      | Lisboa  | 1342664 | 22-04-1989      | 647        |

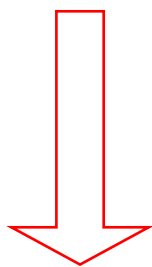


$\sigma$  (Clientes)<sub>Nome like 'Jo%'</sub>

| Nome               | Cidade | BI      | Data_nascimento | N_de_conta |
|--------------------|--------|---------|-----------------|------------|
| Joaquim Alves Dias | Guarda | 1256673 | 18-01-1978      | 556        |
| Joana Antunes      | Lisboa | 1022634 | 23-07-1988      | 801        |

# Operações mais básicas: PROJEÇÃO $\pi$

| Nome               | Cidade  | BI      | Data_nascimento | N_de_conta |
|--------------------|---------|---------|-----------------|------------|
| António Silva      | Coimbra | 1234343 | 12-03-1968      | 900        |
| Joaquim Alves Dias | Guarda  | 1256673 | 18-01-1978      | 556        |
| Maria Teresa Horta | Lisboa  | 1275432 | 15-09-1978      | 900        |
| Joana Antunes      | Lisboa  | 1022634 | 23-07-1988      | 801        |
| Luísa Saraiva      | Lisboa  | 1342664 | 22-04-1989      | 647        |



$\pi$ (Clientes)<sub>Nome, BI</sub>

| Nome               | BI      |
|--------------------|---------|
| António Silva      | 1234343 |
| Joaquim Alves Dias | 1256673 |
| Maria Teresa Horta | 1275432 |
| Joana Antunes      | 1022634 |
| Luísa Saraiva      | 1342664 |

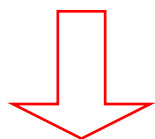
# Operações mais básicas: INTERSECÇÃO $\cap$

Clientes

| Nome               | Cidade  | BI      | Data_nascimento | N_de_conta |
|--------------------|---------|---------|-----------------|------------|
| António Silva      | Coimbra | 1234343 | 12-03-1968      | 900        |
| Joaquim Alves Dias | Guarda  | 1256673 | 18-01-1978      | 556        |
| Maria Teresa Horta | Lisboa  | 1275432 | 15-09-1978      | 900        |
| Joana Antunes      | Lisboa  | 1022634 | 23-07-1988      | 801        |
| Luísa Saraiva      | Lisboa  | 1342664 | 22-04-1989      | 647        |

Clientes1

| Nome               | Cidade  | BI      | Data_nascimento | N_de_conta |
|--------------------|---------|---------|-----------------|------------|
| António Silva      | Coimbra | 1234343 | 12-03-1968      | 900        |
| Joaquim Alves Dias | Guarda  | 1256673 | 18-01-1978      | 556        |



Clientes  $\cap$  Clientes1

| Nome               | Cidade  | BI      | Data_nascimento | N_de_conta |
|--------------------|---------|---------|-----------------|------------|
| António Silva      | Coimbra | 1234343 | 12-03-1968      | 900        |
| Joaquim Alves Dias | Guarda  | 1256673 | 18-01-1978      | 556        |

UC

bd

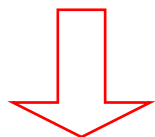
# Operações mais básicas: **UNIÃO U**

Clientes

| Nome               | Cidade  | BI      | Data_nascimento | N_de_conta |
|--------------------|---------|---------|-----------------|------------|
| António Silva      | Coimbra | 1234343 | 12-03-1968      | 900        |
| Joaquim Alves Dias | Guarda  | 1256673 | 18-01-1978      | 556        |
| Maria Teresa Horta | Lisboa  | 1275432 | 15-09-1978      | 900        |

Clientes1

| Nome               | Cidade  | BI      | Data_nascimento | N_de_conta |
|--------------------|---------|---------|-----------------|------------|
| António Silva      | Coimbra | 1234343 | 12-03-1968      | 900        |
| Joaquim Alves Dias | Guarda  | 1256673 | 18-01-1978      | 556        |



Clientes **U** Clientes1

| Nome               | Cidade  | BI      | Data_nascimento | N_de_conta |
|--------------------|---------|---------|-----------------|------------|
| António Silva      | Coimbra | 1234343 | 12-03-1968      | 900        |
| Joaquim Alves Dias | Guarda  | 1256673 | 18-01-1978      | 556        |
| Maria Teresa Horta | Lisboa  | 1275432 | 15-09-1978      | 900        |

UC

bd

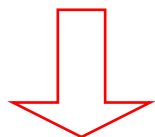
# Operações básicas: **Diferença** \

Clientes

| Nome               | Cidade  | BI      | Data_nascimento | N_de_conta |
|--------------------|---------|---------|-----------------|------------|
| António Silva      | Coimbra | 1234343 | 12-03-1968      | 900        |
| Joaquim Alves Dias | Guarda  | 1256673 | 18-01-1978      | 556        |
| Maria Teresa Horta | Lisboa  | 1275432 | 15-09-1978      | 900        |

Clientes1

| Nome               | Cidade  | BI      | Data_nascimento | N_de_conta |
|--------------------|---------|---------|-----------------|------------|
| António Silva      | Coimbra | 1234343 | 12-03-1968      | 900        |
| Joaquim Alves Dias | Guarda  | 1256673 | 18-01-1978      | 556        |



Clientes \ Clientes1

| Nome               | Cidade | BI      | Data_nascimento | N_de_conta |
|--------------------|--------|---------|-----------------|------------|
| Maria Teresa Horta | Lisboa | 1275432 | 15-09-1978      | 900        |



# Agregação – group by - having

Select função, ndep, avg(14\*sal), sum(sal)  
From emp  
Group by função, ndep

$\gamma_{\text{função, ndep}} (\text{função, ndep, avg(14*sal), sum(sal)}) (\text{emp})$

$\pi_{\text{função, ndep, avg(14*sal), sum(sal)}} \{$   
 $\quad (\gamma_{\text{função, ndep}} (\text{função, ndep, avg(14*sal), sum(sal)}) (\text{emp}))$   
 $\quad \}$

# Agregação – group by - having

```
Select função, ndep, avg(14*sal), sum(sal)
From emp
Group by função, ndep
Having sum(sal)<20000
```

$\pi_{\text{função, ndep, avg}(14*\text{sal}), \text{sum}(\text{sal})} \{$

$H_{\text{sum}(\text{sal}) < 20000} (\gamma_{\text{função, ndep}} (\text{função, ndep, avg}(14*\text{sal}), \text{sum}(\text{sal})) (\text{emp}))$   
 $\}$

# Equivalent in Relational Algebra

```
SELECT emp.nome, sal, funcao, dep.nome  
FROM emp, dep  
WHERE  
emp.ndep=dep.ndep AND ndep = 10 AND sal > 200;
```

UC

$$\pi_{\text{emp.nome, sal, funcao, dep.nome}} [\sigma_{\text{emp.sal} > 200} (\sigma_{\text{dep.ndep} = 10} (\text{emp} \bowtie_{\text{ndep} = \text{ndep}} \text{dep}))]$$

bd

# Write equivalent in Relational Algebra

```
SELECT emp.nome, sal, funcao, dep.nome  
FROM emp, dep  
WHERE  
emp.ndep=dep.ndep AND ndep = 10 AND sal > 200;
```

# FAZER: Equivalente em Algebra Relacional

```
SELECT emp.nome, escalao, dep.nome  
FROM emp, dep, descontos  
WHERE  
emp.ndep=dep.ndep AND dep.nome='Contabilidade' AND sal <250  
and sal between salinf and salsup;
```

UC

```
SELECT emp.nome, funcao, sal  
FROM emp INTERSECT (select * from emp where nemp<3000)  
where sal < 250  
and funcao='Contabilista';
```

E como simplificar = otimizar esta ultima pesquisa?

bd

# RESPOSTA: Equivalente em Algebra Relacional

```
SELECT emp.nome, escalao, dep.nome
FROM emp, dep, descontos
WHERE
emp.ndep=dep.ndep AND dep.nome='Contabilidade' AND sal <250
and sal between salinf and salsup;
```

$$\pi_{\text{emp.nome, escalao, dep.nome}} \left( \sigma_{\text{dep.nome='Contabilidade'}} \left( \sigma_{\text{sal} < 250} \left( \text{dep} \bowtie_{\text{emp.ndep=dep.ndep}} \left( \text{emp} \bowtie_{\text{sal between salinf and salsup}} \text{descontos} \right) \right) \right) \right)$$

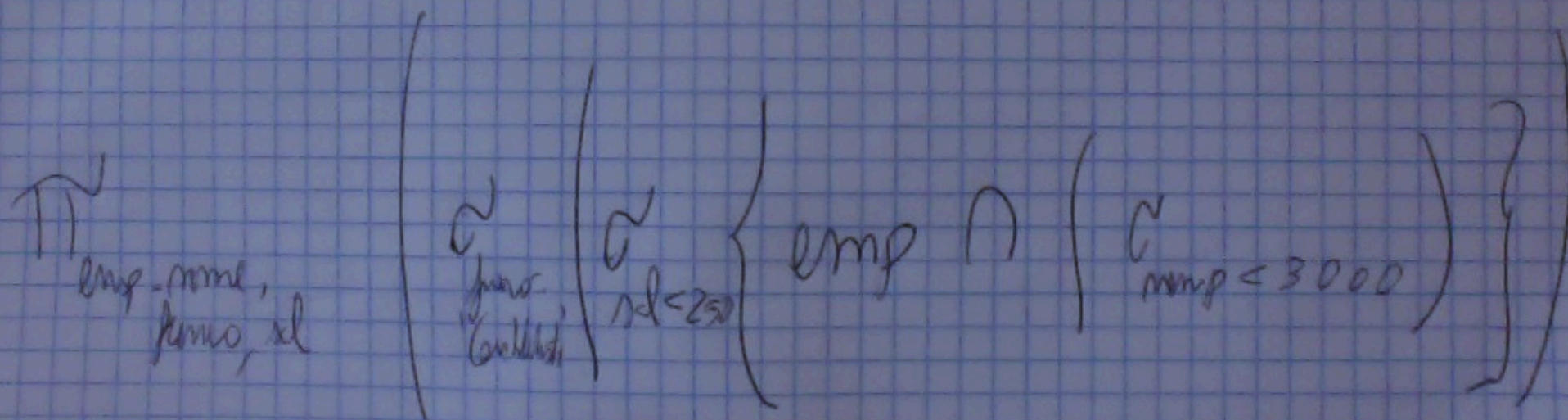
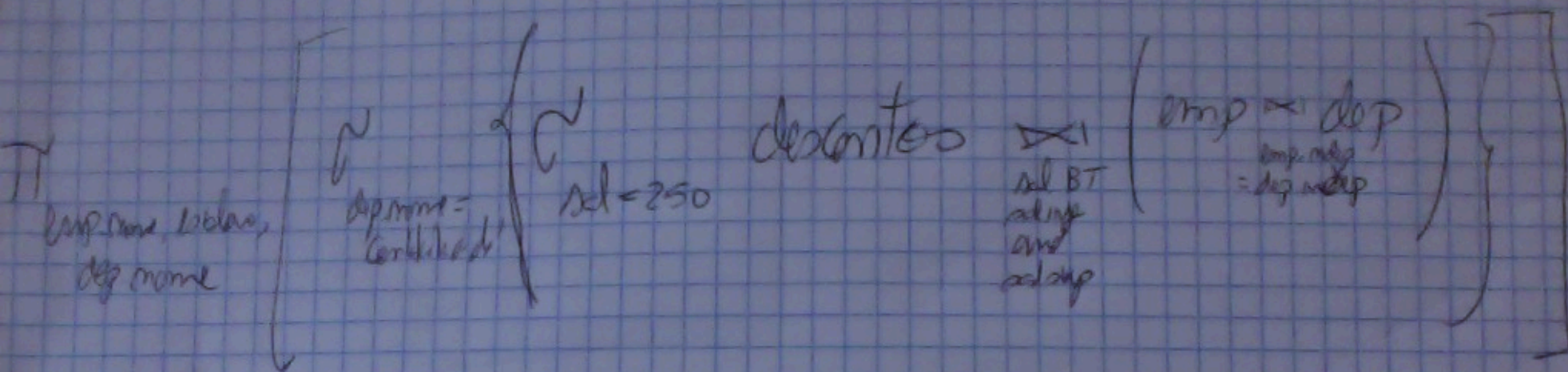
UC

```
SELECT emp.nome, funcao, sal
FROM emp INTERSECT (select * from emp where nemp<3000)
where sal < 250
and funcao='Contabilista';
```

E como simplificar = otimizar esta pesquisa?

bd

```
SELECT emp.nome, dep.nome, avg(sal)
FROM emp JOIN dep ON
(emp.ndep=dep.ndep)
where sal < 250
Group by emp.nome, dep.nome
```





Dúvidas?

**FIM**