

Banco de Dados

Bacharelado em Ciência da Computação

Prof. Dr. Ronaldo Celso Messias Correia
ronaldo.correia@unesp.br

Álgebra Relacional

- Modelo de Dados:
 - definição de restrições e estruturas da base de dados
 - conjunto de operações para manipular dados
 - Modelo Relacional => **Álgebra Relacional**

- Linguagens de Consulta Formais
 - Duas linguagens de consulta matemáticas formam a base para a definição e para a implementação de linguagens “reais” (isto é, SQL):
 - Álgebra relacional: mais operacional, útil para representar planos de execução de consultas
 - Cálculo relacional: permite que o usuário especifique o que deseja, sem dizer como o sistema deve proceder.
 - Conhecimentos de álgebra relacional são fundamentais para se entender SQL!

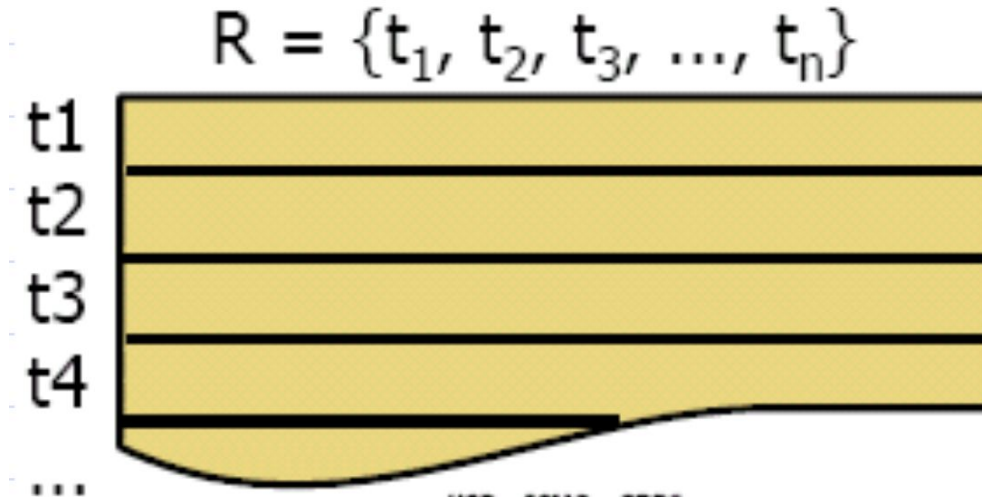
Álgebra Relacional

- Álgebra Relacional
 - linguagem de consulta procedural
 - composta por um conjunto de operações utilizadas para manipular relações
- Operação da Álgebra Relacional
 - Definida sobre uma ou mais relações, resultando sempre em uma relação
- Expressão da Álgebra Relacional
 - Sequência de operações
- Operadores de Comparação
 - $=$, $<$, \leq , $>$, \geq , \neq
- Operadores lógicos
 - \wedge (and), \vee (or), \neg (not)

- Operações sobre conjuntos
 - União
 - União Exclusiva
 - Interseção
 - Diferença
 - Produto Cartesiano
- Operações Relacionais Unárias
 - Seleção
 - Projeção
 - Rename
- Operações Relacionais Binárias
 - Junção
 - Divisão

Operações sobre conjuntos

- Operações usuais da Teoria dos Conjuntos
- Na Álgebra Relacional cada relação é considerada um conjunto de tuplas
- Levam em consideração apenas a estrutura da relação e não a semântica
- Nas operações binárias sobre conjuntos a maioria exige Compatibilidade de Domínio nas relações



Exemplos de Tabelas

Aluno = {CPF, Nome, Idade, Curso}

{<010.225.222-74, Paulo, 25, computação>,
<111.222.333-44, Ana, 18, eletrônica>,
<222.333.444-55, Silvio, 19, odontologia>,
<333.444.555-66, Zenir, 18, computação>}

Professor= {CPF, Nome, Idade, Depto}

{<010.225.222-74, Paulo, 25, computação>,
<110.325.222-74, Aylton, 30, computação>,
<650.455.222-87, Marta, 26, eletrônica>}

Dom(Curso) = Dom(Depto)

Atributo mais adequado:
Data_Nascimento

Operações sobre conjuntos

- União ($R \cup S$) - Tuplas da primeira relação (R) mais tuplas da segunda (S) - tuplas que pertencem à primeira relação ou à segunda.
 - Tomam como entrada duas relações, que precisam ser compatíveis
 - Mesmo número de colunas (atributos)
 - Colunas 'correspondentes' com o mesmo tipo - Domínio
 - Tuplas repetidas são eliminadas
 - Relação resultado tem os nomes dos atributos da primeira relação
 - É possível renomear
 - Operação é comutativa

Operações sobre conjuntos

➤ União

$$\blacksquare \text{ Aluno } \cup \text{ Professor} = \{\text{CPF, Nome, Idade, Curso}\}$$

$\{ \langle 010.225.222-74, \text{Paulo}, 25, \text{computação} \rangle, \\ \langle \text{Ana}, 18, \text{eletrônica} \rangle, \\ \langle \text{Silvio}, 19, \text{odontologia} \rangle, \\ \langle \text{Zenir}, 18, \text{computação} \rangle, \\ \langle \text{Aylton}, 30, \text{computação} \rangle, \\ \langle \text{Marta}, 26, \text{eletrônica} \rangle \}$

Pergunta: Relação de todos que são alunos ou professores?

Resposta: Aluno \cup Professor

Operações sobre conjuntos

- Interseção ($R \cap S$) – Apenas as tuplas que estão na primeira relação (R) e também na segunda (S) - tuplas que pertencem às duas relações.
 - Tomam como entrada duas relações, que precisam ser compatíveis
 - Mesmo número de colunas
 - Colunas 'correspondentes' com o mesmo tipo - Domínio
 - Relação resultado tem os nomes dos atributos da primeira relação
 - É possível renomear
 - Operação é comutativa

Exemplos de Tabelas

Aluno = {CPF, Nome, IdadeA, Curso}

{<010.225.222-74, Paulo, 25, computação>,
<111.222.333-44, Ana, 18, eletrônica>,
<222.333.444-55, Silvio, 19, odontologia>,
<333.444.555-66, Zenir, 18, computação>}

Professor= {CPF, Nome, IdadeP, Depto}

{<010.225.222-74, Paulo, 25, computação>,
<Aylton, 30, computação>,
<Marta, 26, eletrônica>}

➤ Interseção

Pergunta: Relação de todos que são alunos e professores?

- $\text{Aluno} \cap \text{Professor} = \{ \text{CPF, Nome, IdadeA, Curso} \}$
 $\{ \langle 010.225.222-74, \text{Paulo}, 25, \text{computação} \rangle, \}$
- $\text{Professor} \cap \text{Aluno} = \{ \text{CPF, Nome, IdadeP, Depto} \}$
 $\{ \langle 010.225.222-74, \text{Paulo}, 25, \text{computação} \rangle, \}$

Operações sobre conjuntos

- Diferença ($R - S$) – Tuplas que estão na primeira relação (R) mas não estão na segunda (S)
 - Tomam como entrada duas relações, que precisam ser compatíveis
 - Mesmo número de colunas
 - Colunas 'correspondentes' com o mesmo tipo - Domínio
 - Relação resultado tem os nomes dos atributos da primeira relação
 - É possível renomear
 - Operação não comutativa

Exemplos de Tabelas

Aluno = {CPF, Nome, IdadeA, Curso}

{<010.225.222-74, Paulo, 25, computação>,
<111.222.333-44, Ana, 18, eletrônica>,
<222.333.444-55, Silvio, 19, odontologia>,
<333.444.555-66, Zenir, 18, computação>}

Professor= {CPF, Nome, IdadeP, Depto}

{<010.225.222-74, Paulo, 25, computação>,
<Aylton, 30, computação>,
<Marta, 26, eletrônica>}

➤ Diferença

Pergunta: Relação dos alunos que não são professores?

■ Aluno – Professor = { CPF, Nome, IdadeA, Curso}

{<111.222.333-44, Ana, 18, eletrônica>,
<222.333.444-55, Silvio, 19, odontologia>,
<333.444.555-66, Zenir, 18, computação>}

Operações sobre conjuntos

- União Exclusiva ($R \cup | S$) – Tuplas que estão na primeira relação (R) ou na segunda (S), mas não as tuplas que estão em ambas
 - Tomam como entrada duas relações, que precisam ser compatíveis
 - Mesmo número de colunas
 - Colunas 'correspondentes' com o mesmo tipo - Domínio
 - Relação resultado tem os nomes dos atributos da primeira relação
 - É possível renomear
 - Operação comutativa

Exemplos de Tabelas

Aluno = {CPF, Nome, IdadeA, Curso}

{<010.225.222-74, Paulo, 25, computação>,
<111.222.333-44, Ana, 18, eletrônica>,
<222.333.444-55, Silvio, 19, odontologia>,
<333.444.555-66, Zenir, 18, computação>}

Professor= {CPF, Nome, IdadeP, Depto}

{<010.225.222-74, Paulo, 25, computação>,
<012.666.999-85, Aylton, 30, computação>,
<747.654.147-85, Marta, 26, eletrônica>}

➤ União Exclusiva

Pergunta: Relação de todos que são, somente alunos ou somente professores?

■ Aluno \cup Professor = { CPF, Nome, IdadeA, Curso}

{<111.222.333-44, Ana, 18, eletrônica>,
<222.333.444-55, Silvio, 19, odontologia>,
<333.444.555-66, Zenir, 18, computação>,
<012.666.999-85, Aylton, 30, computação>,
<747.654.147-85, Marta, 26, eletrônica>}

Operações sobre conjuntos

- Produto Cartesiano ($R \times S$) – a nova relação tem como atributos a concatenação dos atributos da relação R e da relação S
 - Todas as combinações possíveis de tuplas de R com tuplas de S
 - Relação é definida por um subconjunto de um produto cartesiano de um conjunto de domínios

Operações sobre conjuntos

Produto = {Codpro, NomeP, Preço, CodFor}

{<01, Mouse, 55.00, F-10>,
<02, Teclado, 100.00, F-50>
<03, Monitor, 500.00, F-10>}

Fornecedor = {Codigo, NomeF}

{<F-10, Master>,
<F-50, Kabum>}

➤ Produto X Fornecedor = { Codpro, NomeP, Preço, CodFor, Codigo, NomeF}

Tuplas que tem
algum valores
que podem estar
relacionados

{<01, Mouse, 55.00, F-10, F-10, Master>
<01, Mouse, 55.00, F-10, F-50, Kabum>
<02, Teclado, 100.00, F-50, F-10, Master>
<02, Teclado, 100.00, F-50, F-50, Kabum>
<03, Monitor, 500.00, F-10, F-10, Master>
<03, Monitor, 500.00, F-10, F-50, Kabum>

Operações da Álgebra Relacional

- Assignment (\leftarrow)
 - Atribuição de um nome a uma relação que armazena resultados intermediários de uma expressão algébrica
 - Nome \leftarrow Expressão Algébrica Relacional
 - $A \leftarrow B \cup C$
 - $E \leftarrow A \cup (C \times D)$
 - Renomear os atributos de uma relação intermediária ou resultante de uma sequência de operações
 - NomeRelação(A_1, A_2, \dots) \leftarrow Expressão
 - $A(\text{NovoAtrib1}, \text{NovoAtrib2}) \leftarrow B \cup C$

Operações da Álgebra Relacional

- Rename (ρ – letra minúscula grega rho)
 - Permite renomear uma relação ou os atributos de uma relação
 - Dada uma relação $R (A_1, A_2, \dots, A_n)$
 - $\rho_X (R)$
 - Renomeia a relação R para X
 - $\rho_X (B_1, B_2, \dots, B_n)^{(\text{Expressão})}$
 - Retorna o resultado da expressão sob o nome X , com os atributos recebendo novos nomes

Operações Relacionais Unárias

- Seleção – $\sigma_{(\text{condição})}$ R (letra minúscula grega sigma)
 - Seleciona um subconjunto de tuplas da relação R que satisfazem à condição de seleção
 - A Condição de seleção é sempre uma operação de comparação (=, <, >, etc) de um atributo da relação com:
 - uma constante
 - Com outro atributo da própria relação => comparação de valores de dois atributos da mesma tupla
 - Particionamento horizontal – escolha de algumas linhas (tuplas) da tabela
 - Operador Seleção é Comutativo
 - $\sigma_{(\text{condição A})} R (\sigma_{(\text{condição B})} R) = \sigma_{(\text{condição B})} R (\sigma_{(\text{condição A})} R)$

Operações Relacionais Unárias

Aluno = {Nome, Idade, Curso}

{<Paulo, 25, computação>,
<Ana, 18, eletrônica>,
<Silvio, 19, odontologia>,
<Zenir, 18, computação>}

Seleção

- Selecione os dados dos alunos que cursam eletrônica

➤ $\sigma_{(\text{curso} = \text{"eletrônica"})}$ Aluno
■ Resultado:

<Ana, 18, eletrônica>

A relação resultante pode ser a entrada para outra operação
(composição)

Operações Relacionais Unárias

Aluno = {Nome, Idade, CursoA}

{<Paulo, 25, comp>,
<Ana, 18, eletrônica>,
<Silvio, 19, odontologia>,
<Zenir, 18, estatística>}

Oferece = {CursoO, DeptoO}

{<Comp, DMC>,
<Comp, DF>,
<Matem, DEF>

.....

.....

Seleção: Em quais departamentos cada aluno cumpre as disciplinas de seu curso?

■ AlunoDepto \leftarrow Aluno X Oferece

AlunoDepto = {Nome, Idade, CursoA, CursoO, Depto}

■ $\sigma_{(\text{CursoA} = \text{CursoO})}$ AlunoDepto

■ $\sigma_{(\text{CursoA} = \text{CursoO})}$ (Aluno X Oferece)

{<Paulo, 25, comp, comp, DMC>,
<Paulo, 25, comp, comp, DF>

Aluno = {Nome, Idade, CursoA}

{<Paulo, 25, comp>,
<Ana, 18, eletrônica>,
<Silvio, 19, odontologia>,
<Zenir, 18, estatística>}

Oferece = {CursoO, DeptoO}

{<Comp, DMEC>,
<Comp, DF>,
<Matem, DEF>}

{<Paulo, 25, comp, comp, dmec>,
{<Paulo, 25, comp, comp, df>,
{<Paulo, 25, comp, matem, DEF>,
<Ana, 18, eletrônica, comp,dmec>,
<Ana, 18, eletrônica, comp,df>,
<Ana, 18, eletrônica, matem,def>,
<Silvio, 19, odontologia, comp, dmec>,
<Silvio, 19, odontologia, comp, df>,
<Silvio, 19, odontologia, matem, DEF>,
<Zenir, 18, estatística, comp, dmec>
<Zenir, 18, estatística, comp, DF>
<Zenir, 18, estatística, matem, def>}

select * from aluno, oferece
where CursoA = CursoB

Operações sobre conjuntos

Produto = {Codpro, NomeP, Preço, CodFo, NOmefr}

{<01, Mouse, 55.00, F-10>,
<02, Teclado, 100.00, F-50>
<03, Monitor, 500.00, F-10>}

Fornecedor = {Codigo, NomeF}

{<F-10, Master>,
<F-50, Kabum>}

Pergunta: Qual o respectivo nome do fornecedor de cada produto.

- Produto X Fornecedor = { Codpro, NomeP, Preço, CodFor, Codigo, NomeF}
- $\sigma_{(CodFor = Codigo)}(\text{Produto X Fornecedor})$
- {<01, Mouse, 55.00, F-10, F-10, Master>
<01, Mouse, 55.00, F-10, F-50, Kabum>
<02, Teclado, 100.00, F-50, F-10, Master>
<02, Teclado, 100.00, F-50, F-50, Kabum>
<03, Monitor, 500.00, F-10, F-10, Master>
<03, Monitor, 500.00, F-10, F-50, Kabum>

Operações sobre conjuntos

Produto = {Codpro, NomeP, Preço, CodFor}

{<01, Mouse, 55.00, F-10>,
<02, Teclado, 100.00, F-50>
<03, Monitor, 500.00, F-10>}

Fornecedor = {Codigo, NomeF}

{<F-10, Master>,
<F-50, Kabum>}

Pergunta: Qual o respectivo nome do fornecedor de cada produto.

$\sigma_{(\text{CodFor} = \text{Codigo})}(\text{Produto X Fornecedor})$

{ Codpro, NomeP, Preço, CodFor, Codigo, NomeF}
 <01, Mouse, 55.00, F-10, F-10, Master>
 <02, Teclado, 100.00, F-50, F-50, Kabum>
 <03, Monitor, 500.00, F-10, F-10, Master>

Operações Relacionais Unárias

- **Projeção** – $\pi_{(\text{atributos})} R$ (letra grega pi)
 - Seleciona um subconjunto do conjunto de atributos da relação R, indicados na lista de atributos
 - O resultado de uma operação de projeção é uma relação
 - Não devem existir tuplas repetidas – quaisquer linhas em duplicidades são eliminadas
 - Lista de <atributos> contém a chave da relação - resultado não tem tuplas repetidas
 - Lista de <atributos> não contém a chave – possibilidade de tuplas repetidas
 - Particionamento vertical – escolha de algumas colunas (atributos) da tabela
 - Operador de Projeção não é Comutativo

Operações Relacionais Unárias

Aluno = {Nome, Idade, CursoA}
{<Paulo, 25, comp>,
<Ana, 18, eletrônica>,
<Silvio, 19, odontologia>,
<Zenir, 18, estatística>}

Oferece = {CursoO, DeptoO}
{<Comp, DMC>,
<Comp, DF>,
<Matem, DEF>}

Projeção: Selecione nomes dos alunos e dos departamentos em que cada aluno cumpre as disciplinas de seu curso

Resultado $\leftarrow \pi_{(\text{Nome}, \text{Depto})} \sigma_{(\text{CursoA} = \text{CursoB})} (\text{Aluno} \times \text{Oferece})$

{<Paulo, DMC>,
<Paulo, DF>}

<Paulo, 25, comp, comp, dmec>,
<Paulo, 25, comp, comp, df>}

Operações sobre conjuntos

Produto = {Codpro, NomeP, Preço, CodFor}

{<01, Mouse, 55.00, F-10>,
<02, Teclado, 100.00, F-50>
<03, Monitor, 500.00, F-10>}

Fornecedor = {Codigo, NomeF}

{<F-10, Master>,
<F-50, Kabum>}

Pergunta: Selecione o nome do produto e o nome do fornecedor do produto de código = 01.

$\sigma_{(\text{CodFor} = \text{Codigo}) \wedge (\text{codpro} = 01)} (\text{Produto} \times \text{Fornecedor})$

{ Codpro, NomeP, Preço, CodFor, Codigo, NomeF}
<01, Mouse, 55.00, F-10, F-10, Master>

$\pi_{(\text{nomep}, \text{nomef})} \sigma_{(\text{CodFor} = \text{Codigo}) \wedge (\text{codpro} = 01)} (\text{Produto} \times \text{Fornecedor})$

{ NomeP, NomeF}
<Mouse, Master>

Exercício

Aluno = {RA, Nome, Idade, CursoA}
{<105, Paulo, 25, computação>,
<201, Ana, 18, eletrônica>,
<303, Silvio, 19, odontologia>,
<55, Zenir, 18, estatística>}

Matricula = {RA, Disciplina, Nota}
{<105, 5010, 7.0>,
<105, 5015, 8.5>,
<303, 5010, 6.5>
<55, 5015, 9.5>}

“Listar as disciplinas em que os alunos de computação se matricularam”

$\pi_{(disciplina)} (\sigma_{(cursoA = "computacao")} (\sigma_{(Matricula.RA=aluno.RA)} (Matricula \times Aluno)))$

<https://dbis-uibk.github.io/relax>

Exercício

```
Aluno = {RA, Nome, Idade, CursoA, RA, Disciplina, Nota}  
{<105, Paulo, 25, computação, 105, 5010, 7.0>  
<105, Paulo, 25, computação, 105, 5015, 8.5>  
<105, Paulo, 25, computação, 303, 5010, 6.5>  
<105, Paulo, 25, computação, 55, 5015, 9.5>  
<201, Ana, 18, eletrônica, 105, 5010, 7.0>  
...  
<303, Silvio, 19, odontologia, 105, 5010, 7.0>  
....  
<55, Zenir, 18, estatística, 105, 5010, 7.0>  
....}
```

“Listar as disciplinas em que os alunos de computação se matricularam”

$$\pi_{(\text{disciplina})}(\sigma_{(\text{curso} = \text{"computacao"})}(\sigma_{(\text{Matricula.RA} = \text{aluno.RA})}(\text{Matricula X Aluno})))$$

2 **1**

Operações Relacionais Unárias



Junção (Join)


$$R \bowtie_{\text{cond}} S = \sigma_{\text{cond}}(R \times S)$$

- Usada para combinar tuplas (relacionadas) de diferentes relações em uma única
- Permite combinar certas seleções e um produto cartesiano dentro de uma operação
- Permite processamento de relacionamentos entre relações
- Pode ser computado mais eficientemente que o produto cartesiano

Operações Relacionais Unárias

- Junção Theta (θ -join) - uma operação Junção com uma condição de junção geral
- θ é qualquer operador válido no domínio dos atributos de junção
 - Atributos de junção aparecem ambos na relação resultado (pares de atributos com valores idênticos)

$\pi_{(disciplina)} (\sigma_{(curso = \text{"computacao"})} (\sigma_{(Nome=NomeA)} (Matricula \times Aluno)))$

(Matricula  Aluno)
(Nome = NomeA)

(Matricula [X] (Nome = NomeA) Aluno)

Variações de Junção

- Equi-Junção (Equi-join)
 - θ é um operador de igualdade
 - Os atributos de junção aparecem ambos na relação resultado
 - Duas maneiras de simbolizar

$(R \bowtie_{\text{AtribR} = \text{AtribS}} S)$

$(R \bowtie_{\text{AtribR}, \text{AtribS}} S)$

Variações de Junção

- Junção Natural ($R * S$)
 - Semelhante à equi-junção
 - Apenas um dos atributos de junção aparecem na relação resultado
 - Requer que os atributos de junção tenham nomes iguais nas duas relações ou sejam renomeados

Empregado * Departamento

```
select * from empregado inner join departamento on  
empregado.codep = departamento.codep
```

Variações de Junção

Empregado = {NomeE, Depto}
{<Pedro, D1>,
<Rose, D1>,
<Carlos, D2>}

Departamento = {NomeD, Depto}
{<DMEC, D1>,
<FÍSICA, D2>
<GEO, D3>}

Atributos de junção (Depto) tenham nomes iguais nas duas relações

Empregado * Departamento

{NomeE, Depto, NomeD}
{<Pedro, D1, DMEC>,
<Rose, D1, DMEC>,
<Carlos, D2, FÍSICA>}

Variações de Junção

Empregado = {NomeE, Depto}
{<Pedro, D1>,
<Rose, D1>,
<Carlos, D2>}

Departamento = {NomeD, Codigo}
{<DMEC, D1>,
<FÍSICA, D2>
<GEO, D3>}

Atributos de junção com nomes diferentes nas duas relações - requer renomear

Empregado * P (Codigo, Depto) Departamento

{NomeE, Depto, NomeD}
{<Pedro, D1, DMEC>,
<Rose, D1, DMEC>,
<Carlos, D2, FÍSICA>}

Exercícios

Funcionários = {Numf, Nomef}
{<01, F1>,
<02, F2>,
<03, F3>,
<04, F4>}

Dependentes = {Numf, Nomed, parentesco}
{<01, Alice, filha>,
<02, Alice, esposa>,
<02, Clara, filha>
<03, José, filho>}

1 - Quais os nomes e parentescos de todos os dependentes?

$\pi_{(nomed, parentesco)}$
(dependentes)

2 – Quais funcionários (numf) possuem dependentes filhas?

$\pi_{(numf)}$ $\sigma_{(parentesco=$
filha)

3 – Quais funcionários não possuem dependentes?

$\pi_{(numf)}$ funcionarios - $\pi_{(numf)}$ dependentes

Exercícios

Funcionários = {Numf, Nomef}
{<01, F1>,
<02, F2>,
<03, F3>,
<04, F4>}

Dependentes = {Numf, Nomed, par}
{<01, Alice, filha>,
<02, Alice, esposa>,
<02, Clara, filha>
<03, José, filho>}

4 – Dê os nomes dos funcionários que possuem algum dependente?

Projeção_(Nomef) Seleção_(F.Numf=D.Numf) (Funcionarios X Dependentes)

5 – Dê o nome de cada funcionário que possui uma dependente chamada Alice?

Projeção_(nomef) Seleção_(f.numf = d.numf) F X Seleção_(Nomed = Alice) D

6 – Quais funcionários não tem Alice como dependente (isto é, nenhuma dependente chamada Alice)?

Projeção_(numf) funcionarios - projecao_(numf) selecao_(nomed=Alice) Dependentes