

The slide features five light purple circles arranged in two rows. The top row contains three circles, and the bottom row contains two circles. The text 'Projeto de Banco de Dados' is centered over the top row, and 'Normalização' is positioned to the right of the bottom row.

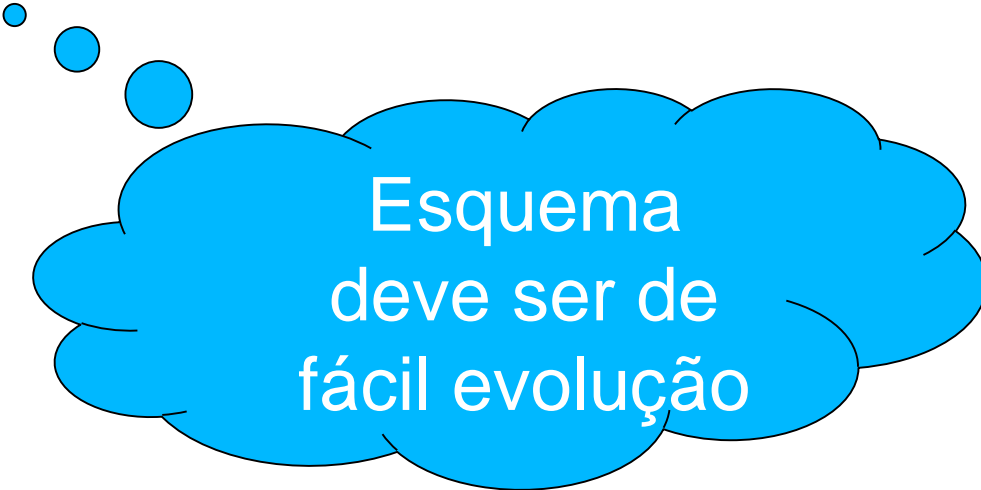
Projeto de Banco de Dados

Normalização

Problemas com projeto de BD

cliente(codigo,nome,endereco)

- Se fosse necessário usar o sobrenome do cliente separado?
- Mesma pergunta para endereço?



Esquema
deve ser de
fácil evolução



Problemas com projeto de BD

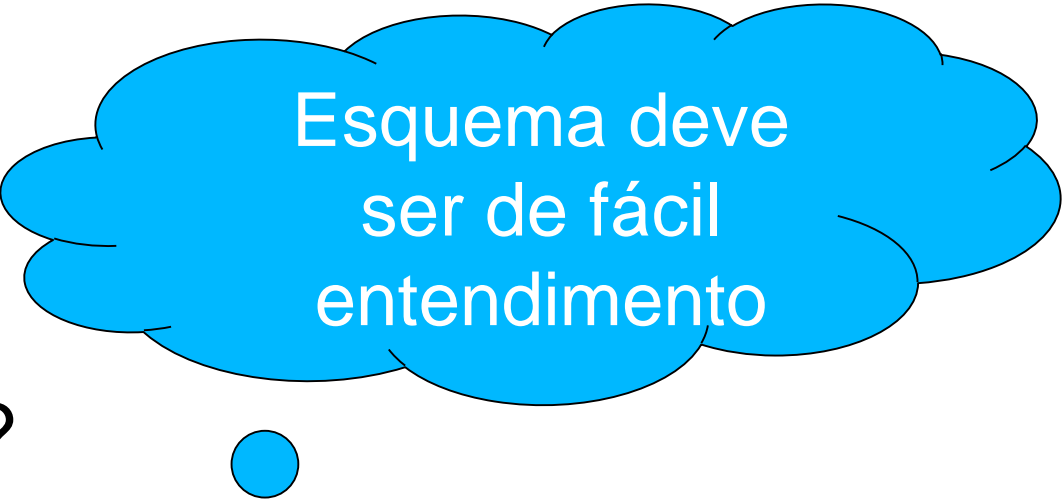
cliente(cid,ncompleto,local)

- O que é cid?
- O que é ncompleto?
- O que é local ?

Problemas com projeto de BD

cliente(cd,nm,rua,cid,end)

- O que é nm?
- O que é cd?
- O que é cid e rua?
- end é o endereço?
 - cid/rua e end não são redundantes?



Esquema deve
ser de fácil
entendimento



Problemas com projeto de BD

- Semântica dos identificadores
 - Nomes de tabelas, atributos, restrições
- Campos nulos
 - Podem gerar respostas inesperadas em algumas consultas
- Anomalias de operações
 - Inserção, Exclusão e Alteração

Problemas com projeto de BD

- Cliente(idC, nomeC, ruaC, cidadeC)
 - EmprestimoCliente(idC, numEm, quantia)
 - (1, Joao, Rua A, São Paulo)
 - (2, Paulo, Rua B, Rio de Janeiro)
-
- (1, E-001, 20000)
 - (2, E-001, 20000)
- quantia é redundante neste esquema
-

Problemas com projeto de BD

- Cliente(idC,nomeC,ruaC,cidadeC)
- EmprestimoCliente(idC,numEm)
- Emprestimo(numEmp, quantia)
- (1,Joao,Rua A,São Paulo)
- (2, Paulo, Rua B, Rio de Janeiro)
- (1,E-001)
- (2, E-001)
- (E-001,20000)

Exemplos de esquemas com problemas

(a) EMP_DEPT

ENOME	<u>SSN</u>	DATANASC	ENDERECO	DNUMERO	DNOME	DGERSSN
-------	------------	----------	----------	---------	-------	---------

(b) EMP_PROJ

<u>SSN</u>	<u>PNUMERO</u>	HORAS	ENOME	PNOME	PLOCALIZACAO
------------	----------------	-------	-------	-------	--------------

EMP_DEPT					redundância	
ENOME	SSN	DATANASC	ENDERECO	DNUMERO	DNOME	DGERSSN
Smith,John B.	123456789	1965-01-09	731 Fondren,Houston,TX	5	Pesquisa	333445555
Wong,Franklin T.	333445555	1955-12-08	638 Voss,Houston,TX	5	Pesquisa	333445555
Zelaya, Alicia J.	999887777	1968-07-19	3321 Castle,Spring,TX	4	Administracao	987654321
Wallace,Jennifer S.	987654321	1941-06-20	291 Berry,Bellaire,TX	4	Administracao	987654321
Narayan,Ramesh K.	666884444	1962-09-15	975 FireOak,Humble,TX	5	Pesquisa	333445555
English,Joyce A.	453453453	1972-07-31	5631 Rice,Houston,TX	5	Pesquisa	333445555
Jabbar,Ahmad V.	987987987	1969-03-29	980 Dallas,Houston,TX	4	Administracao	987654321
Borg,James E.	888665555	1937-11-10	450 Stone,Houston,TX	1	Sede Administrativa	888665555

redundância			redundância		
SSN	PNUMERO	HORAS	ENOME	PNOME	PLOCALIZACAO
123456789	1	32.5	Smith,John B.	ProdutoX	Bellaire
123456789	2	7.5	Smith,John B.	ProdutoY	Sugarland
666884444	3	40.0	Narayan,Ramesh K.	ProdutoZ	Houston
453453453	1	20.0	English,Joyce A.	ProdutoX	Bellaire
453453453	2	20.0	English,Joyce A.	ProdutoY	Sugarland
333445555	2	10.0	Wong,Franklin T.	ProdutoY	Sugarland
333445555	3	10.0	Wong,Franklin T.	ProdutoZ	Houston
333445555	10	10.0	Wong,Franklin T.	Automação	Stafford
333445555	20	10.0	Wong,Franklin T.	Reorganização	Houston
999887777	30	30.0	Zelaya,Alicia J.	NovosBenefícios	Stafford
999887777	10	10.0	Zelaya,Alicia J.	Automação	Stafford
987987987	10	35.0	Jabbar,Ahmad V.	Automação	Stafford
987987987	30	5.0	Jabbar,Ahmad V.	Novos benefícios	Stafford
987654321	30	20.0	Wallace,Jennifer S.	Novos benefícios	Stafford
987654321	20	15.0	Wallace,Jennifer S.	Reorganização	Houston
888665555	20	null	Borg,James E.	Reorganização	Houston

Processo Formal para um bom projeto de BD

- Normalização

- Processo de análise de determinados esquemas de relações para
 - Reagrupar informações de forma a minimizar redundâncias
 - Regrupar informações de forma a minimizar anomalias
 - Inserção, atualização e exclusão
- 5 formas normais
 - Quanto maior a forma normal do esquema, mais decomposto é o banco de dados (mais tabelas)

Primeira forma normal

- Domínio atômico
 - Elementos são considerados unidades indivisíveis
- Um esquema relacional R está na primeira forma normal se os domínios de todos os atributos de R forem atômicos
 - Atributos não podem ser compostos
 - Atributos não podem ser multivalorados

Primeira Forma Normal

- Esquema Cliente não está na primeira forma normal
 - Cliente (cod, nome, (telefone))
- Passagem para 1FN
 - Eliminar tabelas aninhadas (atributos multivalorados)
 - Chave primária da tabela original deve compor a tabela aninhada
 - Novos esquemas estão na 1FN
 - Cliente(cod,nome)
 - Telefone(cod,telefone)
cod referencia Cliente(cod)

Normalização (Livro Heuser)

Usada para Engenharia Reversa de Arquivos

RELATÓRIO DE ALOCAÇÃO A PROJETO

CÓDIGO DO PROJETO: LSC001

TIPO: Novo Desenv.

DESCRIÇÃO: Sistema de Estoque

<i>CÓDIGO DO EMPREGADO</i>	<i>NOME</i>	<i>CATEGORIA FUNCIONAL</i>	<i>SALÁRIO</i>	<i>DATA DE INÍCIO NO PROJETO</i>	<i>TEMPO ALOCADO AO PROJETO</i>
2146	João	A1	4	1/11/91	24
3145	Sílvio	A2	4	2/10/91	24
6126	José	B1	9	3/10/92	18
1214	Carlos	A2	4	4/10/92	18
8191	Mário	A1	4	1/11/92	12

CÓDIGO DO PROJETO: PAG02

TIPO: Manutenção

DESCRIÇÃO: Sistema de RH

<i>CÓDIGO DO EMPREGADO</i>	<i>NOME</i>	<i>CATEGORIA FUNCIONAL</i>	<i>SALÁRIO</i>	<i>DATA DE INÍCIO NO PROJETO</i>	<i>TEMPO ALOCADO AO PROJETO</i>
8191	Mário	A1	4	1/05/93	12
4112	João	A2	4	4/01/91	24
6126	José	B1	9	1/11/92	12

Normalização

Notação do livro Carlos Alberto Heuser

Representação do arquivo anterior em formato de tabela

CódProj	Tipo	Descr	Emp					
			CodEmp	Nome	Cat	Sal	DataIni	TempAl
LSC001	Novo Desenv.	Sistema de Estoque	2146	João	A1	4	1/11/91	24
			3145	Silvio	A2	4	2/10/91	24
			6126	José	B1	9	3/10/92	18
			1214	Carlos	A2	4	4/10/92	18
			8191	Mário	A1	4	1/11/92	12
PAG02	Manutenção	Sistema de RH	8191	Mário	A1	4	1/05/93	12
			4112	João	A2	4	4/01/91	24
			6126	José	B1	9	1/11/92	12

Representação do arquivo na notação do modelo relacional

Proj (CodProj, Tipo, Descr,
(CodEmp, Nome, Cat, Sal, DataIni, TempAl))

Primeira Forma Normal

CódProj	Tipo	Descr	Emp					
			CodEmp	Nome	Cat	Sal	DataIni	TempAl
LSC001	Novo Desenv.	Sistema de Estoque	2146	João	A1	4	1/11/91	24
			3145	Sílvio	A2	4	2/10/91	24
			6126	José	B1	9	3/10/92	18
			1214	Carlos	A2	4	4/10/92	18
			8191	Mário	A1	4	1/11/92	12
PAG02	Manutenção	Sistema de RH	8191	Mário	A1	4	1/05/93	12
			4112	João	A2	4	4/01/91	24
			6126	José	B1	9	1/11/92	12

Proj (CodProj, Tipo, Descr,
(CodEmp, Nome, Cat, Sal, DataIni, TempAl))

**Esquema Proj não está na primeira forma normal: a coluna “Emp”
é multivalorada**

Emp é uma tabela aninhada

Primeira Forma Normal

**Proj e ProjEmp
estão na 1FN**

Proj:

CódProj	Tipo	Descr
LSC001	Novo Desenv.	Sistema de Estoque
PAG02	Manutenção	Sistema de RH

ProjEmp:

CódProj	CodEmp	Nome	Cat	Sal	DataIni	TempAl
LSC001	2146	João	A1	4	1/11/91	24
LSC001	3145	Sílvio	A2	4	2/10/91	24
LSC001	6126	José	B1	9	3/10/92	18
LSC001	1214	Carlos	A2	4	4/10/92	18
LSC001	8191	Mário	A1	4	1/11/92	12
PAG02	8191	Mário	A1	4	1/05/93	12
PAG02	4112	João	A2	4	4/01/91	24
PAG02	6126	José	B1	9	1/11/92	12

Dependência Funcional

- Seja R um esquema de relação

$$\mathbf{a} \subseteq R \text{ e } \mathbf{b} \subseteq R$$

- A Dependência Funcional

$$\mathbf{a} \rightarrow \mathbf{b} \text{ (se } \mathbf{a} \text{ então } \mathbf{b})$$

O atributo \mathbf{b} é funcionalmente dependente de \mathbf{a} se cada valor da coluna \mathbf{a} determina um e somente um valor da coluna \mathbf{b}

Dependência Funcional

- Seja R um esquema de relação

$$\alpha \subseteq R \text{ e } \beta \subseteq R$$

- A Dependência Funcional

$$\alpha \rightarrow \beta \text{ (se } \alpha \text{ então } \beta)$$

se aplica em R se e somente se para quaisquer relações r , sempre que quaisquer duas tuplas t_1 e t_2 de r concordarem nos atributos α , elas também concordam nos atributos β .

$$\text{Ou seja, } t_1[\alpha] = t_2[\alpha] \Rightarrow t_1[\beta] = t_2[\beta]$$

Dependência Funcional

parte dominante da DF

$a \rightarrow b$

- Em uma tabela T , uma coluna **b** depende de uma coluna **a** (coluna a determina b), quando em todas as linhas da tabela, para cada valor da coluna **a** , aparece o mesmo valor na coluna **b**

Codigo \rightarrow Salario

...	Código	...	Salário	...
	E1		10	
	E3		10	
	E1		10	
	E2		5	
	E3		10	
	E2		5	
	E1		10	

Dependência Funcional

- Considere $r(A,B,C,D)$ com a seguinte instância de r

A	B	C	D
A1	B1	C1	D1
A1	B2	C1	D2
A2	B2	C2	D2
A2	B3	C2	D3
A3	B3	C2	D4

$A \rightarrow C$

~~**$C \rightarrow A$**~~

~~**$C \rightarrow D$**~~

Dependência Funcional

A	B	C	D
B	5	2	20
C	4	2	15
B	6	7	20
B	5	2	20
C	2	2	15
C	4	2	15
A	10	5	18
A	12	3	18
A	10	5	18
B	5	2	20
C	4	2	15
A	10	5	18
C	4	2	15

Dependências funcionais:

$(A,B) \rightarrow C$

$A \rightarrow D$

Dependência Funcional

(a) EMP_DEPT

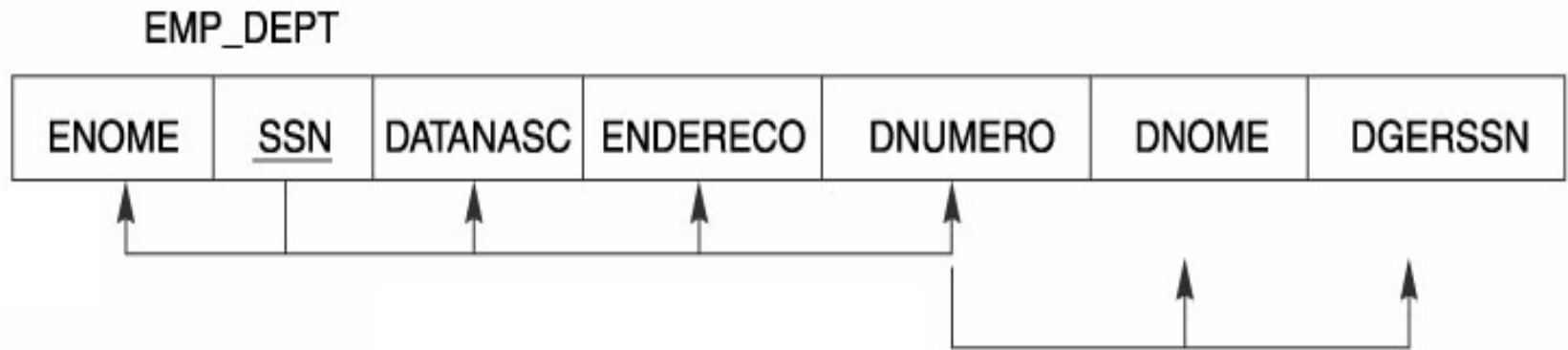
ENOME	<u>SSN</u>	DATANASC	ENDERECO	DNUMERO	DNOME	DGERSSN
-------	------------	----------	----------	---------	-------	---------

(b) EMP_PROJ

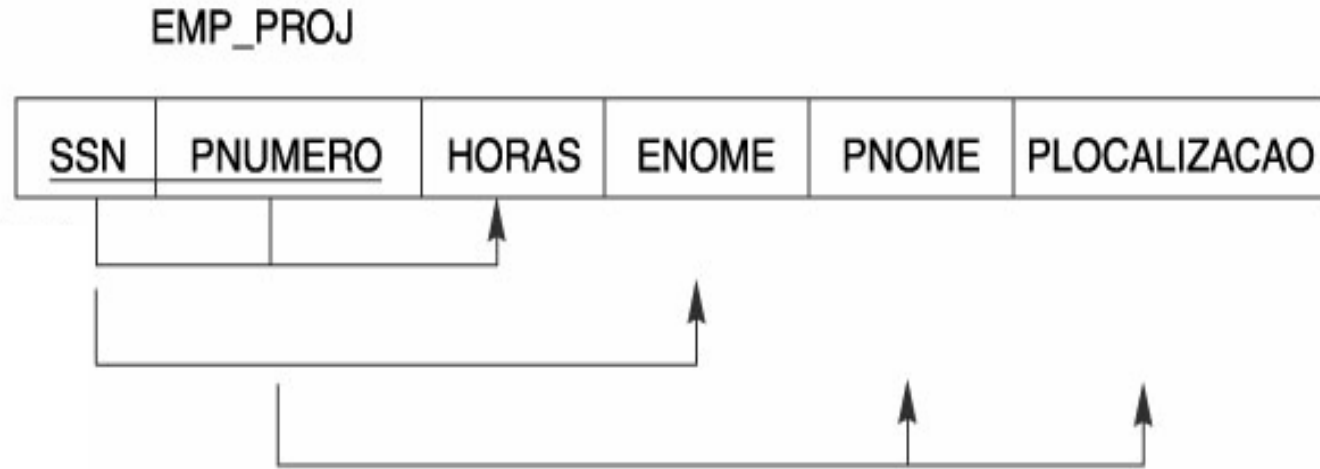
<u>SSN</u>	<u>PNUMERO</u>	HORAS	ENOME	PNOME	PLOCALIZACAO
------------	----------------	-------	-------	-------	--------------

Dependência Funcional (representação no livro Elmasri e Navathe)

(a)



(b)



Definições

- Quando todos os atributos em uma relação dependem de um conjunto de atributos, podemos dizer que este conjunto é uma chave candidata
 - Trabalho(codTrab, codProj, codEmp, horas)
 - $\text{codProj}, \text{codEmp} \rightarrow \text{horas}$ e $\text{codProj}, \text{codEmp} \rightarrow \text{codTrab}$
 - $\text{codProj}, \text{codEmp}$ é uma chave candidata
 - Cliente(codigo, cpf, nome, endereco)
 - $\text{codigo} \rightarrow \text{cpf}, \text{nome}, \text{endereco}$
 - $\text{cpf} \rightarrow \text{codigo}, \text{nome}, \text{endereco}$
 - cpf é uma chave candidata

Dependência Funcional

- Dependência funcional total e parcial

<u>SSN</u>	<u>PNUMERO</u>	HORAS	ENOME	PNOME	PLOCALIZACAO
------------	----------------	-------	-------	-------	--------------

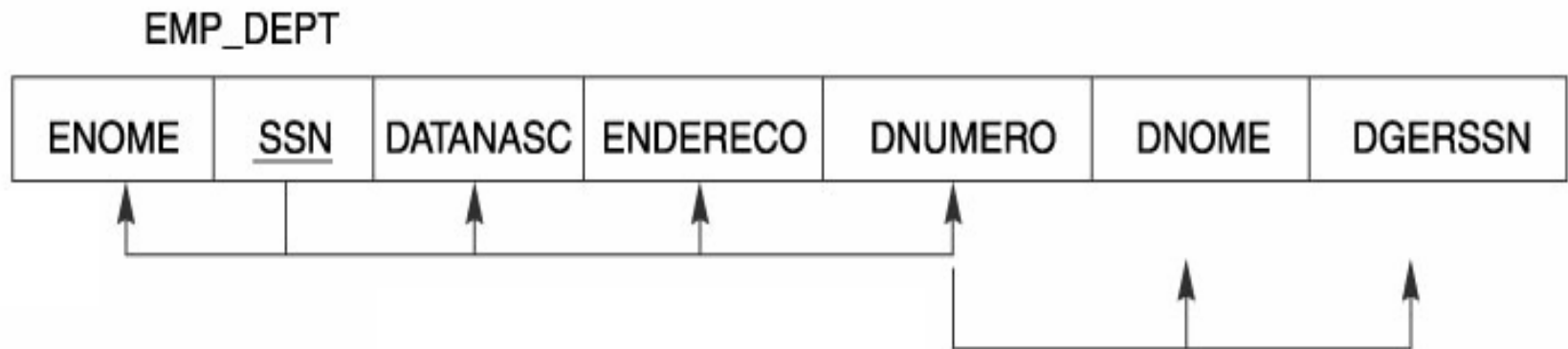
- coluna horas depende totalmente da parte dominante
 - $ssn, pnumero \rightarrow horas$ **(DF total)**
- coluna pnome depende somente de pnumero (não depende totalmente da parte dominante)
 - $ssn, pnumero \rightarrow pnome$ **(DF parcial)**

Segunda forma normal – 2FN

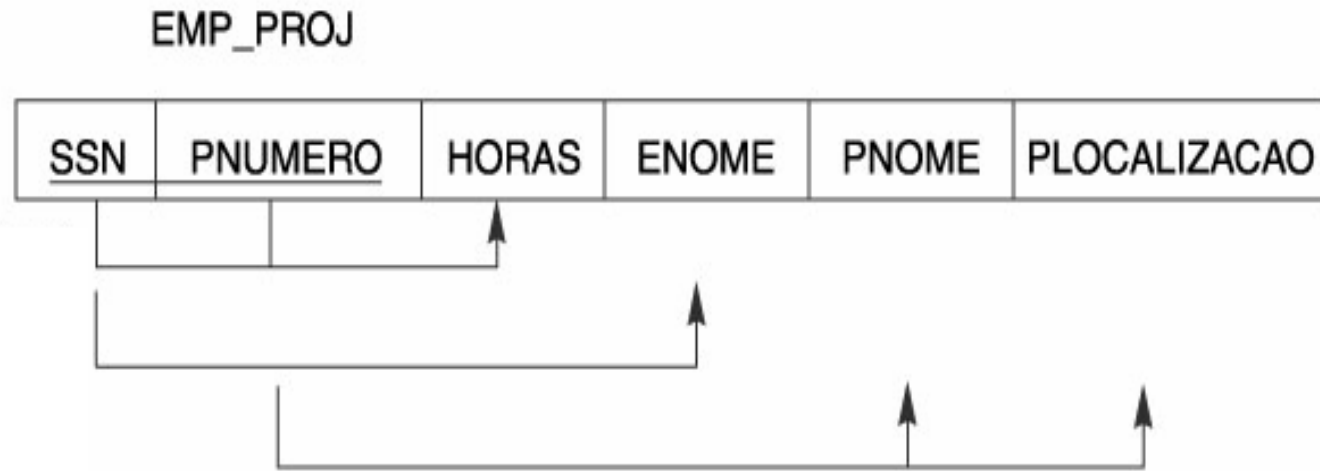
- Um esquema de relação R está na 2FN se todo atributo que não é chave primária tem dependência funcional total da chave primária
- Toda coluna que não é chave primária deve depender totalmente da chave primária

Analise o esquema quanto à 2FN

(a)



(b)



Segunda Forma Normal

(a) EMP_DEPT Está na segunda forma normal

ENOME	<u>SSN</u>	DATANASC	ENDERECO	DNUMERO	DNOME	DGERSSN
-------	------------	----------	----------	---------	-------	---------

(b) EMP_PROJ NÃO está na segunda forma normal

<u>SSN</u>	<u>PNUMERO</u>	HORAS	ENOME	PNOME	PLOCALIZACAO
------------	----------------	-------	-------	-------	--------------

ssn,pnumero → pnome (parcial)

ssn,pnumero → plocalizacao (parcial)

ssn,pnumero → enome (parcial)

Segunda Forma Normal

(a) EMP_DEPT Está na segunda forma normal

ENOME	<u>SSN</u>	DATANASC	ENDERECO	DNUMERO	DNOME	DGERSSN
-------	------------	----------	----------	---------	-------	---------

(b) EMP_PROJ NÃO está na segunda forma normal

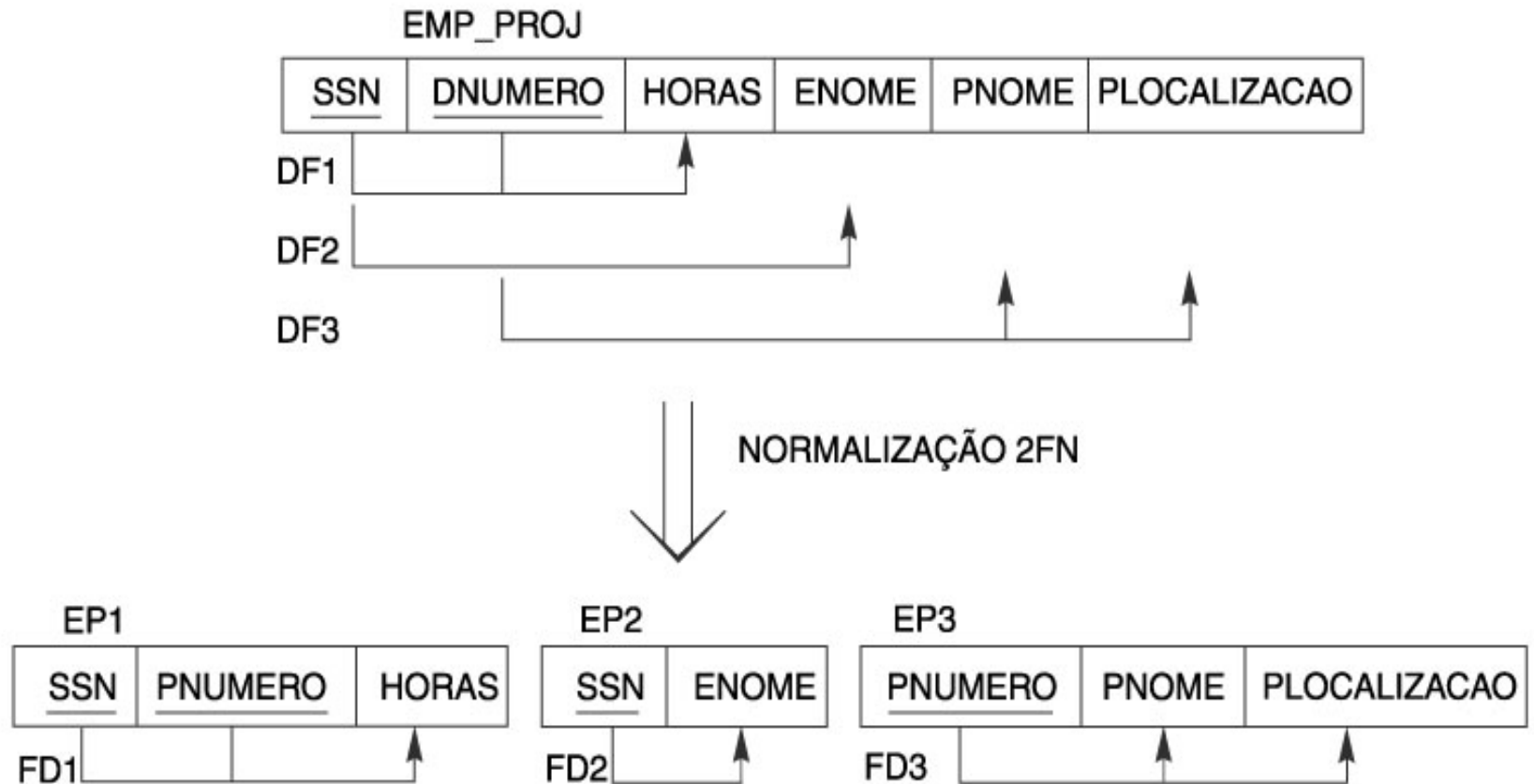
<u>SSN</u>	<u>PNUMERO</u>	HORAS	ENOME	PNOME	PLOCALIZACAO
------------	----------------	-------	-------	-------	--------------

ssn,pnumero → pnome (parcial)/ **pnumero → pnome**

ssn,pnumero → plocalizacao (parcial)/ **pnumero → plocalizacao**

ssn,pnumero → enome (parcial)/ **ssn → nome**

Normalização



Segunda Forma Normal

ProjEmp não está na 2FN

ProjEmp:

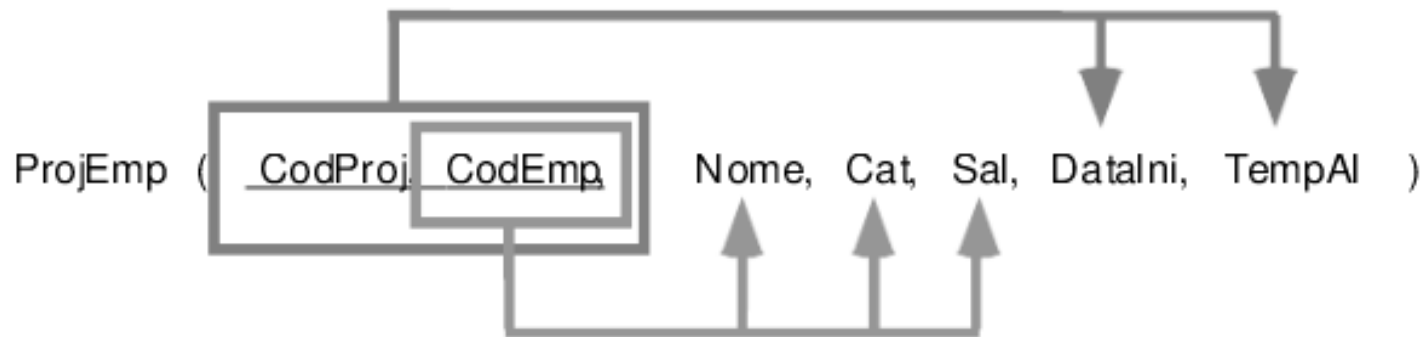
CódProj	CodEmp	Nome	Cat	Sal	DataIni	TempAl
LSC001	2146	João	A1	4	1/11/91	24
LSC001	3145	Sílvio	A2	4	2/10/91	24
LSC001	6126	José	B1	9	3/10/92	18
LSC001	1214	Carlos	A2	4	4/10/92	18
LSC001	8191	Mário	A1	4	1/11/92	12
PAG02	8191	Mário	A1	4	1/05/93	12
PAG02	4112	João	A2	4	4/01/91	24
PAG02	6126	José	B1	9	1/11/92	12

codProj, CodEmp → DataIni (total)

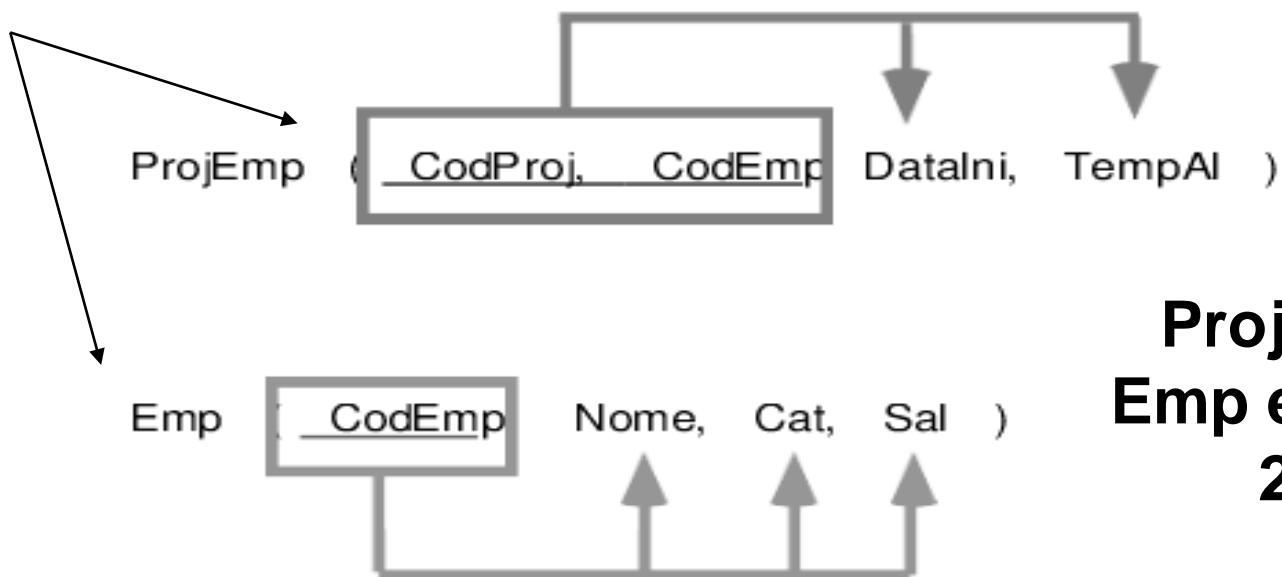
codProj, CodEmp → Nome (parcial)

Segunda Forma Normal

Passagem de ProjEmp para 2FN



Decomposição



**ProjEmp e
Emp estão na
2FN**

Dependência Funcional

- Dependência transitiva

- $X \rightarrow Y$ é transitiva em Z se
- Existir um conjunto de atributos Y que dependa de outro atributo Z que não é chave:
- $X \rightarrow Z$ e $Z \rightarrow Y$

ENOME	<u>SSN</u>	DATANASC	ENDERECO	DNUMERO	DNOME	DGERSSN
-------	------------	----------	----------	---------	-------	---------

$SSN \rightarrow DNOME$ é transitiva em $DNUMERO$

$SSN \rightarrow DNUMERO$ e $DNUMERO \rightarrow DNOME$

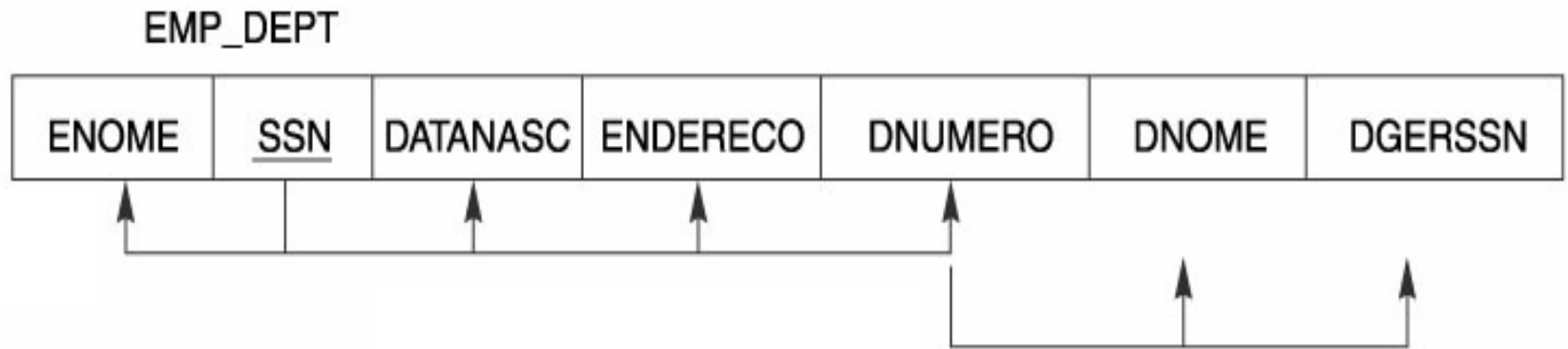
Terceira Forma Normal



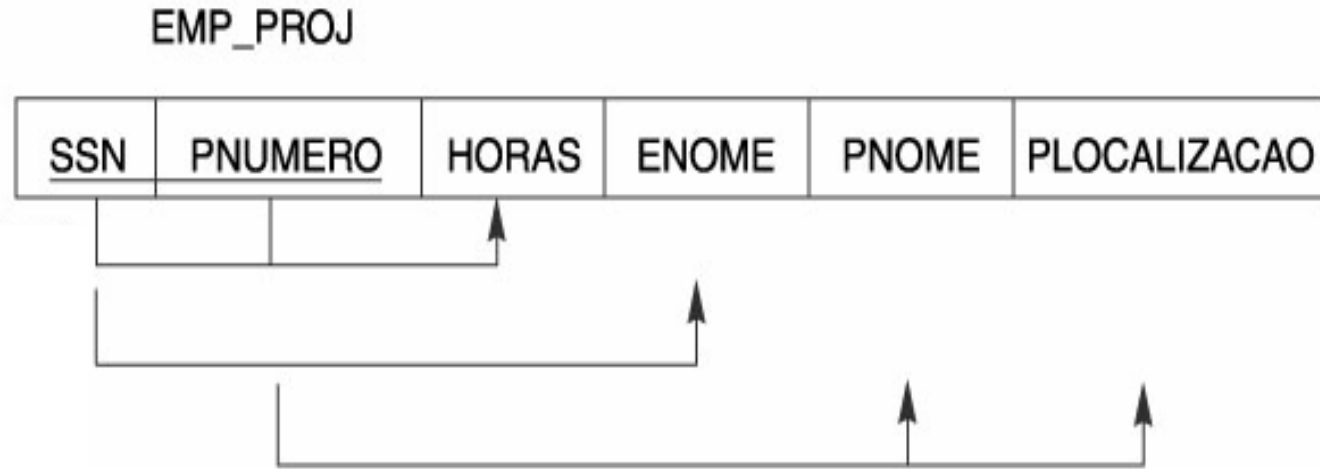
- Um esquema de relação está na 3FN se estiver na 2FN e nenhum atributo não primário for transitivamente dependente da chave primária
OU (em outras palavras)
- Os atributos que não são chave não devem ser funcionalmente dependente de outro atributo que não seja a chave primária

Analise o esquema quanto à 3FN

(a)



(b)

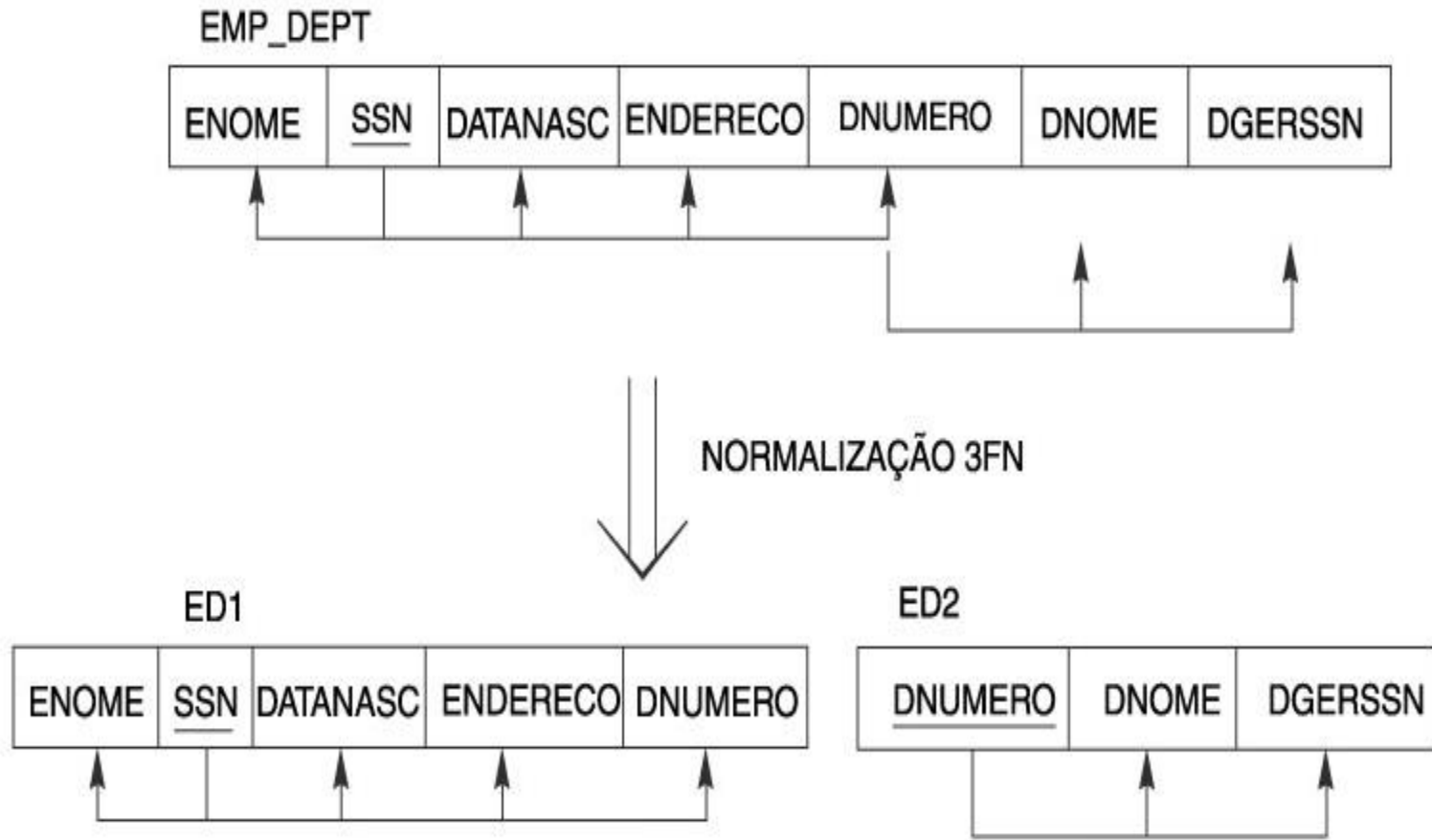


Terceira Forma Normal

ENOME	<u>SSN</u>	DATANASC	ENDERECO	DNUMERO	DNOME	DGERSSN
-------	------------	----------	----------	---------	-------	---------

- Emp_Dept está na segunda forma normal pois nenhum atributo não primário depende parcialmente da chave primária
- Mas não está na 3FN pois DGERSSN e DNOME dependem transitivamente de SSN ou DGERSSN e DNOME dependem de outro atributo que não é a chave

Terceira Forma Normal



Exercício

Exercício 6.2: No contexto de um sistema de controle acadêmico, considera a tabela abaixo:

Matricula

(CodAluno,CodTurma,CodDisciplina,NomeDisciplina,NomeAluno,
CodLocalNascAluno,NomeLocalNascAluno)

As colunas possuem o seguinte significado:

CodAluno - código do aluno matriculado

CodTurma - código da turma na qual o aluno está matriculado (código é o identificador de turma)

CodDisciplina - código que identifica a disciplina da turma

NomeDisciplina - nome de uma disciplina da turma

NomeAluno - nome do aluno matriculado

CodLocalNascAluno - código da localidade em que nasceu o aluno

NomeLocalNascAluno - nome da localidade em que nasceu o aluno

Verifique se a tabela obedece a segunda e a terceira forma normais. Caso não obedeça, faça as transformações necessárias

Exercícios



- Quais são as df?
 - CodAluno -> NomeAluno, CodLocalNascAluno
 - CodLocalNascAluno -> NomeLocalNascAluno
 - CodDisciplina -> NomeDisciplina
 - CodTurma->CodDisciplina

Exercícios

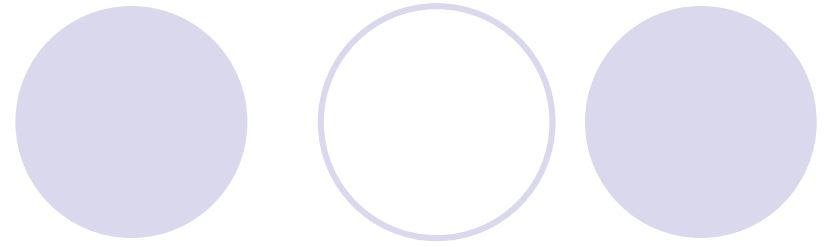
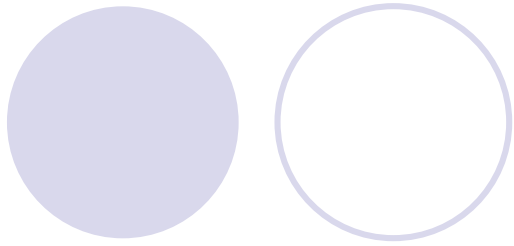
- Quais são as df?
 - CodAluno -> NomeAluno, CodLocalNascAluno, NomeLocalNascAluno
 - CodLocalNascAluno -> NomeLocalNascAluno
 - CodDisciplina -> NomeDisciplina
 - CodTurma->CodDisciplina

**Dependência funcional transitiva
(NomeLocalNascAluno depende de
CodLocalNascAluno que depende
de CodAluno)**

Exercício

(CodAluno, CodTurma, CodDisciplina, NomeDisciplina, NomeAluno,
CodLocalNascAluno, NomeLocalNascAluno)

- Esquema está na 1FN, mas não está na 2FN porque existem atributos que dependem parcialmente da chave primária
 - **CodAluno, CodTurma -> NomeAluno, CodLocalNascAluno, NomeLocalNascAluno**
 - **CodAluno, CodTurma -> CodDisciplina**



Exercício

(CodAluno, CodTurma, CodDisciplina, NomeDisciplina, NomeAluno, CodLocalNascAluno, NomeLocalNascAluno)

- Normalização para 2FN

(CodAluno, NomeAluno, CodLocalNascAluno, NomeLocalNascAluno)

(CodAluno, CodTurma)

(CodTurma, CodDisciplina, NomeDisciplina)

(CodAluno, NomeAluno, CodLocalNascAluno, NomeLocalNascAluno)

(CodAluno, CodTurma)

(CodTurma, CodDisciplina, NomeDisciplina)

- Esquema anterior está na 2FN, mas não está na 3FN porque existem atributos que dependem transitivamente da chave primária
 - CodAluno -> NomeAluno, CodLocalNascAluno
 - CodLocalNascAluno -> NomeLocalNascAluno
 - CodTurma->CodDisciplina
 - CodDisciplina -> NomeDisciplina

Exercício



- Esquema anterior está na 2FN, mas não está na 3FN porque existem atributos que dependem transitivamente da chave primária

- Normalização para 3FN

(CodAluno, NomeAluno, CodLocalNascAluno)

(CodLocalNascAluno, NomeLocalNascAluno)

(CodAluno, CodTurma)

(CodTurma, CodDisciplina)

(CodDisciplina, NomeDisciplina)

Definições

- Atributo primário

- A é um atributo primário em R se A é membro de alguma chave candidata de R

- Superchave

- Qualquer conjunto de atributos que contém uma chave

Definição geral da 2FN

- Um esquema de relação R está na 2FN se cada atributo não primário A tem dependência funcional total de todas as chaves de R

Definição geral da 2FN

- Considere a tabela Trabalho que possui uma chave primária e uma candidata
 - Trabalho(codTrab, codProj, codEmp, horas)
 - codTrab \rightarrow horas, codProj, codEmp
 - codProj, codEmp \rightarrow horas, codTrab
 - Suponha que tenha sido criado um atributo pNome que depende funcionalmente de codProj

Trabalho(codTrab, codProj, codEmp, horas, pNome)
codProj, codEmp \rightarrow pnome é parcial pois
codProj \rightarrow pnome
Logo Trabalho não está na 2FN

Definição geral da 3FN



- Todo atributo não primário de R deve apresentar as seguintes condições
 - Ter dependência funcional total para todas as chaves
 - Não ser dependente transitivamente de nenhuma chave de R

Problemas com as Decomposições

- Maior custo de processamento das consultas (mais junções)
 - Filial(cod,nome,siglaE,nomeE)
 - Dados da filial podem ser consultados juntos com dados do Estado (siglaE e nomeE)
 - Mas a tabela não está 3FN
 - $\text{cod} \rightarrow \text{nome}, \text{siglaE}, \text{nomeE}$ e $\text{siglaE} \rightarrow \text{nomeE}$
 - nomeE depende transitivamente de cod a partir de siglaE (nomeE depende funcionalmente de siglaE que depende funcionalmente de cod)

Problemas com as Decomposições

- Maior custo de processamento das consultas (mais junções)
 - Filial(cod,nome,siglaE,nomeE)
 - cod -> nome,siglaE,nomeE e siglaE ->nomeE
 - Normalização para 3FN (Decomposição 3FN)
 - Filial(cod,nome,siglaE)
 - Estado(siglaE,nomeE)

Problemas com as Decomposições

- Decomposição com perda
 - Quando uma decomposição é mal projetada e não é possível recuperar as instâncias originais do banco antes da decomposição

Filial (cod, nome, siglaE, nomeE)

(1, 'Lojas A', 'MG', 'Minas Gerais')

(2, 'Lojas B', 'RJ', 'Rio de Janeiro')

Decomposição

Filial(cod, nome) Estado(siglaE, nomeE)

(1, 'Lojas A') ('MG', 'Minas Gerais')

(2, 'Lojas B') ('RJ', 'Rio de Janeiro')

Como saber o estado de uma filial?

Problemas com as Decomposições

- Decomposição sem perda
 - Uma decomposição é sem perda quando, para todas as instâncias do banco de dados
 - Seja $R1$ e $R2$ duas tabelas resultantes da decomposição d de R .
 - d é sem perda se
 - Se $(R1 \cap R2)$ for superchave de $R1$ ou $R2$)
 - $R1 \cap R2 \rightarrow R1$ ou
 - $R1 \cap R2 \rightarrow R2$
 - d é uma decomposição sem perda

Problemas com as Decomposições

- Decomposição com perda

Filial (cod, nome, siglaE, nomeE)

Decomposição d1

Filial(cod, nome) e Estado(siglaE, nomeE)

$Filial \cap Estado = \emptyset$, logo $Filial \cap Estado$ não é superchave de nenhuma das tabelas e d1 é com perda

- Decomposição sem perda

Filial (cod, nome, siglaE, nomeE)

Decomposição d2

Filial(cod, nome, siglaE) e Estado(siglaE, nomeE)

$Filial \cap Estado = siglaE$, logo $Filial \cap Estado$ é superchave de Estado e d2 é sem perda

Problemas com as Decomposições

- Decomposição com perda

Filial (cod, nome, siglaE, nomeE)

Decomposição d1

Filial(cod, nome) e Estado(siglaE, nomeE)

$Filial \cap Estado = \emptyset$, logo $Filial \cap Estado$ não é superchave de nenhuma das tabelas e d1 é com perda

- Decomposição sem perda

Filial (cod, nome, siglaE, nomeE)

Decomposição d2

Filial(cod, nome, siglaE) e Estado(siglaE, nomeE)

$Filial \cap Estado = siglaE$, logo $Filial \cap Estado$ é superchave de Estado e d2 é sem perda

Exercício: Identifique as dependências funcionais - Sistema acadêmico

- O sistema deve gerenciar as informações dos Alunos (nome, identidade, matricula, endereco, telefone, dataNasc). A identidade e matricula podem ser usadas, separadamente, como identificadores do aluno.
- Para os departamentos, o sistema deve permitir o gerenciamento das seguintes informações: código, nome e telefone. Tanto o código, quanto o nome podem ser usados como identificadores do departamento
- Cada curso tem um nome, descrição, numero, cargaHoraria e o departamento (Código). O número do curso é diferente para cada curso.
- Cada disciplina tem um nome, o tutor, o ano, semestre, um curso (codigo) e um número. O número da disciplina identifica unicamente uma disciplina.
- Cada aluno tem uma nota em uma disciplina. Um aluno pode fazer várias disciplinas e uma disciplina pode ser ministrada para vários alunos.

Exercício: Identifique as dependências funcionais - Sistema acadêmico

- O sistema deve gerenciar as informações dos Alunos (nome, identidade, matricula, endereco, telefone, dataNasc). A identidade e matricula podem ser usadas, separadamente, como identificadores do aluno.
- Para os departamentos, o sistema deve permitir o gerenciamento das seguintes informações: código, nome e telefone. Tanto o código, quanto o nome podem ser usados como identificadores do departamento
- Cada curso tem um nome, descrição, numero, cargaHoraria e o departamento (Código). O número do curso é diferente para cada curso.
- Cada disciplina tem um nome, o tutor, o ano, semestre, um curso (codigo) e um número. **O número é usado para diferenciar disciplinas diferentes do mesmo curso que são oferecidas no mesmo semestre do mesmo ano.**
- Cada aluno tem uma nota em uma disciplina. Um aluno pode fazer várias disciplinas e uma disciplina pode ser ministrada para vários alunos.

Exercício – Normalize a tabela criada a partir do arquivo abaixo:

CodConcurso: C001 Data: 01/03/2010

Contratante: B01, Nome Contratante: Banco do Brasil

Resultado

CD01	Joao Paulo	P1	Matemática	9
		P2	Literatura	8
CD02	Maria	P1	Matemática	10
		P2	Literatura	7

CodConcurso: C002 Data: 05/04/2012

Contratante: B02, Nome Contratante: Banco do Itáu

Resultado

CD01	Joao Paulo	P1	Matemática	10
		P2	Literatura	9
CD03	Lucia	P1	Matemática	10

Concurso(codConcurso,dataConcurso,codContratante,nomeContratante,

(codCandidato,nomeCandidato,

(codProva,nomeProva, nota)))