

NORMALIZAÇÃO DE BANCOS DE DADOS

Sergio Mergen

Baseado em: ©Carlos A. Heuser - Transparências para uso com o livro
Projeto de Banco de Dados, Ed. Sagra&Luzzatto, Porto Alegre, 1999

Normalização de dados

- Consiste em criar modelos de dados livres de redundância
- Problemas em ter dados redundantes
 - Múltiplos pontos de mudança aumentam as chances de inconsistências
 - Perda de eficiência na atualização

Exemplo de redundâncias

Proj

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

ProjEmp

CódProj	CodEmp	Nome	Cat	Sal	DataIni	TempAI
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

Exemplo de redundâncias

Proj

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

ProjEmp

CódProj	CodEmp	Nome	Cat	Sal	DataIni	TempAI
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

Forma normal

- Regra que uma tabela deve obedecer para ser considerada livre de algum tipo de redundância
- Há diversas formas normais, cada vez mais rígidas
- Inicialmente tratadas
 - primeira forma normal (1FN)
 - segunda forma normal (2FN)
 - terceira forma normal (3FN)

Primeira forma normal - 1FN

primeira forma normal (1FN)

=

diz-se que uma tabela está na primeira forma normal, quando ela não contém tabelas aninhadas

Tabelas aninhadas

- Tabelas aninhadas podem aparecer quando se está criando um banco de dados a partir de documentos de entrada
 - relatórios
 - arquivos de código fonte
 - telas de sistemas legados
 - ...

Tabelas aninhadas - exemplo

RELATÓRIO DE ALOCAÇÃO A PROJETO

CÓDIGO DO PROJETO: LSC001

TIPO: Novo Desenv.

DESCRIÇÃO: Sistema de Estoque

CÓDIGO DO EMPREGADO	NOME	CATEGORIA FUNCIONAL	SALÁRIO	DATA DE INÍCIO NO PROJETO	TEMPO ALOCADO AO PROJETO
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

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

Tabelas aninhadas

- O documento do slide anterior pode ser visualizado como uma tabela com tabelas aninhadas

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	Sílvia	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

Tabelas aninhadas

- O esquema para essa tabela pode ser descrito da seguinte forma

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

- Perceba a existência de tabelas aninhadas
- Esse modelo não está na primeira forma normal
 - Ele é **ÑN**

Passagem à 1FN - alternativas

- Duas alternativas
 - Construir uma única tabela com redundância de dados
 - Construir uma tabela para cada tabela aninhada

Passagem à 1FN - uma única tabela

- Uma tabela na qual os dados das linhas externas à tabela aninhada são repetidos para cada linha da tabela aninhada

Tabela ÑN

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

Tabela 1FN

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

Dados do projeto aparecem repetidos para cada empregado do projeto

Passagem à 1FN – várias tabelas

- Cria-se uma tabela referente a própria tabela que está sendo normalizada e uma tabela para cada tabela aninhada

Tabela ÑN

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

Tabelas 1FN

Proj (CodProj, Tipo, Descr)

ProjEmp (CodProj, CodEmp, Nome, Cat, Sal, DataIni, TempAl)

codProj referencia Proj

Passagem à 1FN - alternativas

- Primeira alternativa (tabela única) é mais correta
- Decompor uma tabela em várias tabelas (segunda alternativa)
 - podem ser perdidas relações entre informações
- Para fins didáticos, veremos a segunda alternativa (várias tabelas)
 - Quando houver diversas tabelas aninhadas, fica difícil visualizar a tabela na 1FN se for usada uma única tabela

Passagem à 1FN - passo 1

- Criar uma tabela na 1FN referente à tabela mais externa
- A chave primária da tabela na 1FN é idêntica à chave da tabela mais externa

ÑÑ

ProjEmp (codProj, Tipo, Descr,
(CodEmp, Nome Cat, Sal, DataIni, TempoAl))

1FN

Proj (codProj, Tipo, Descr)

Passagem à 1FN - Passo 2

- Para cada tabela aninhada
 - criar uma tabela na 1FN composta pelas seguintes colunas:
 - a chave primária de cada uma das tabelas externas na qual a tabela em questão está aninhada
 - Que serão trazidas como chaves estrangeiras
 - as colunas da própria tabela aninhada

ÑÑ

ProjEmp (codProj, Tipo, Descr,
(CodEmp, Nome Cat, Sal, DataIni, TempoAl))



1FN

ProjEmp (codProj, CodEmp, Nome Cat, Sal, DataIni, TempoAl)

codProj referencia Proj

Proj (codProj, Tipo, Descr)

Passagem à 1FN – Passo 2

- Resta definir a chave primária para a tabela retirada do aninhamento

1FN

Proj (codProj, Tipo, Descr)

ProjEmp(CodProj, CodEmp, Nome Cat, Sal, DataIni, TempoAl)

codProj referencia Proj



Pergunta a fazer:

“um valor de **CodEmp** (chave da tabela origem) aparece uma vez só no documento ou várias?”

Passagem à 1FN – Passo 3

- Resta definir a chave primária para a tabela retirada do aninhamento

1FN

Proj (codProj, Tipo, Descr)

ProjEmp(CodProj, CodEmp, Nome Cat, Sal, DataIni, TempoAl)

codProj referencia Proj



Caso cada empregado trabalhe em um único projeto apenas, ou seja, caso um valor de CodEmp apareça uma única vez na tabela

Esse não é o caso

Passagem à 1FN – Passo 3

- Resta definir a chave primária para a tabela retirada do aninhamento

1FN

Proj (codProj, Tipo, Descr)

ProjEmp(CodProj, CodEmp, Nome Cat, Sal, DataIni, TempoAl)

codProj referencia Proj



Como um valor de **CodEmp** aparece várias vezes, é necessário **CodProj** para distinguir as várias aparições

Esse é o modelo final na primeira forma normal

Tabelas na 1FN

Proj

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

ProjEmp

CódProj	CodEmp	Nome	Cat	Sal	DataIni	TempAI
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

Exemplo

ÑN

Arq-Candidatos (Cod-Curso, Nome-Curso, Numero-Vagas-Curso,
(Cod-Cand, Nome-Cand, Escore-Cand))

Exemplo

ÑN

Arq-Candidatos (Cod-Curso, Nome-Curso, Numero-Vagas-Curso,
(Cod-Cand, Nome-Cand, Escore-Cand))



1FN

Cursos

(Cod-Curso, Nome-Curso, Numero-Vagas-Curso)

Exemplo

ÑN

Arq-Candidatos (Cod-Curso, Nome-Curso, Numero-Vagas-Curso,
(Cod-Cand, Nome-Cand, Escore-Cand))



1FN Cursos (Cod-Curso, Nome-Curso, Numero-Vagas-Curso)
Candidatos (Cod-Curso, Cod-Cand, Nome-Cand, Escore-Cand)
codCurso referencia Curso

Exemplo

ÑN

Arq-Candidatos (Cod-Curso, Nome-Curso, Numero-Vagas-Curso,
(Cod-Cand, Nome-Cand, Escore-Cand))



1FN

Cursos (Cod-Curso, Nome-Curso, Numero-Vagas-Curso)

Candidatos (Cod-Curso, Cod-Cand, Nome-Cand, Escore-Cand)

codCurso referencia Curso

Qual é a chave primária da tabela candidatos?

Pergunta a fazer:

“um valor de **Cod-Cand** (chave da tabela origem) aparece uma só vez no documento ou várias?”

Exemplo

ÑN

Arq-Candidatos (Cod-Curso, Nome-Curso, Numero-Vagas-Curso,
(Cod-Cand, Nome-Cand, Escore-Cand))



1FN

Cursos (Cod-Curso, Nome-Curso, Numero-Vagas-Curso)
Candidatos (Cod-Curso, Cod-Cand, Nome-Cand, Escore-Cand)
codCurso referencia Curso

Um código de candidato aparece uma única vez

DEPENDÊNCIA FUNCIONAL

Dependência funcional

- Para entender 2FN e 3FN
 - é necessário compreender o conceito de *dependência funcional*.
- Em uma tabela relacional, diz-se que
 - uma coluna C_2 *depende funcionalmente* de uma coluna C_1 (ou que a coluna C_1 *determina* a coluna C_2) quando
 - em todas linhas da tabela, para cada valor de C_1 que aparece na tabela, aparece o mesmo valor de C_2 .
- Notação usada para indicar que uma coluna C_2 *depende funcionalmente* de uma coluna C_1

$$C_1 \rightarrow C_2$$

Exemplo de dependência funcional

CódProj	CodEmp	Nome	Cat	Sal	DataIni	TempAl
	2146	João				
	3145	Sílvio				
	6126	José				
	2146	João				
	8191	Mário				
	8191	Mário				
	4112	João				
	6126	José				

CodEmp → Nome

Exemplo de dependência funcional

CódProj	CodEmp	Nome	Cat	Sal	DataIni	TempAl
	2146	João				
	3145	Sílvio				
	6126	José				
	2146	João				
	8191	Mário				
	8191	Mário				
	4112	João				
	6126	José				

Nome \nrightarrow CodEmp

Exemplo de dependência funcional

CódProj	CodEmp	Nome	Cat	Sal	DataIni	TempAl
LSC001						24
LSC001						24
LSC001						18
LSC001						18
LSC001						12
PAG02						12
PAG02						24
PAG02						12

codProj ~~→~~ tempoAl

Exemplo de dependência funcional

CódProj	CodEmp	Nome	Cat	Sal	DataIni	TempAl
LSC001	2146					24
LSC001	3145					24
LSC001	6126					18
LSC001	1214					18
LSC001	8191					12
PAG02	8191					12
PAG02	4112					24
PAG02	6126					12

codProj,codEmp → tempoAl

SEGUNDA FORMA NORMAL

Segunda forma normal - 2FN

- Onde está a redundância aqui?

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

Segunda forma normal - 2FN

- Onde está a redundância aqui?

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

Segunda forma normal - 2FN

- Exemplo de situação indesejada:
 - A troca do nome de um funcionário precisa ser feita em vários registros

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 M.	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 M.	B1	9	1/11/92	12

Segunda forma normal - 2FN

segunda forma normal (2FN)

=

uma tabela encontra-se na segunda forma normal,
quando, além de estar na 1FN, não contém
dependências parciais

Dependência parcial

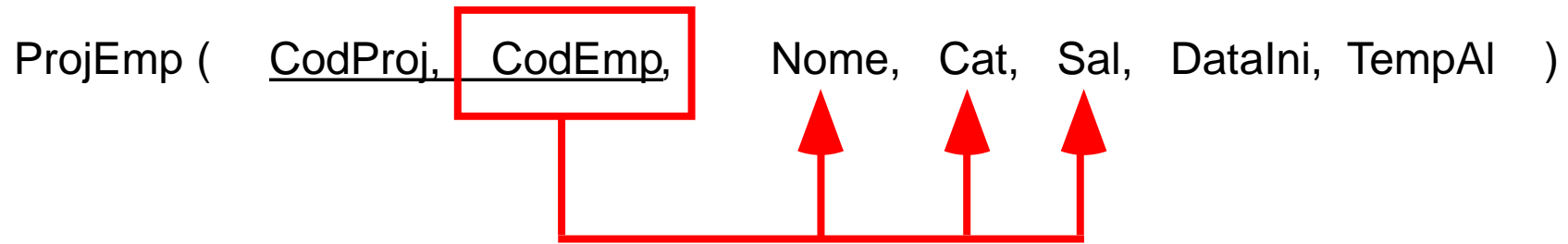
dependência parcial

=

uma dependência (funcional) parcial ocorre quando uma coluna depende apenas *de parte de* uma chave primária composta

Dependências parciais

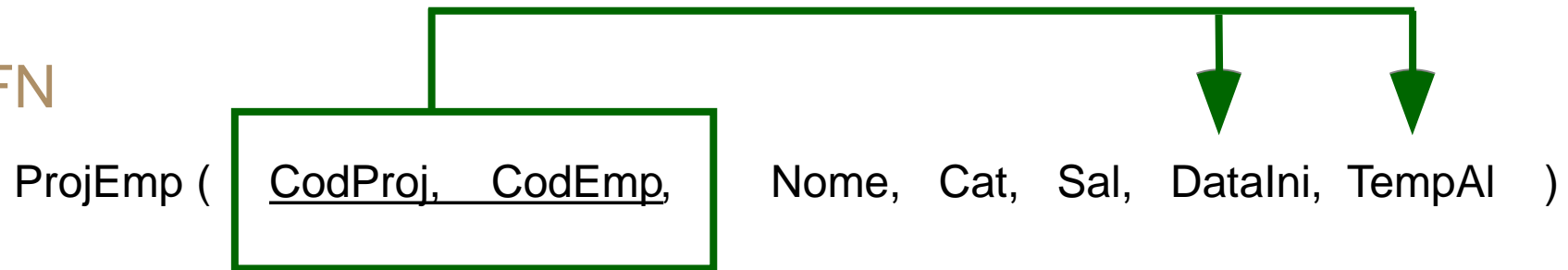
1FN



- Dependência funcional existente
 - codEmp → nome, cat, sal

Dependências não parciais

1FN



- Dependência funcional existente
 - $\text{codProj, codEmp} \rightarrow \text{dataIni, tempoAL}$

Passagem à 2FN

Tabela com uma coluna na chave

1FN

Proj (codProj, Tipo, Descr)

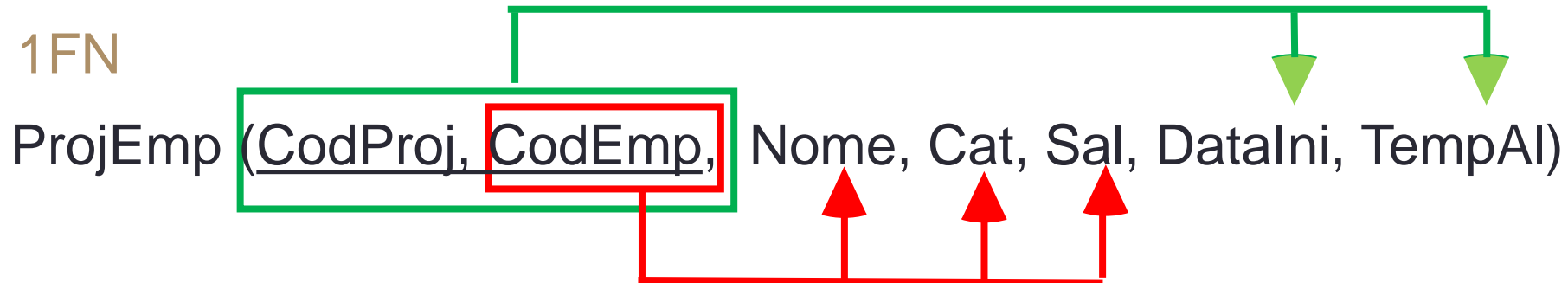


2FN

Proj (codProj, Tipo, Descr)

Passagem à 2FN

Tabela com múltiplas colunas na chave



2FN

ProjEmp (CodProj, CodEmp, DataIni, TempAI)

CodEmp referencia Emp

Emp (CodEmp, Nome, Cat, Sal)

Tabelas na 2FN

Proj

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

ProjEmp

CódProj	CodEmp	DataIni	TempAI
LSC001	2146	1/11/91	24
LSC001	3145	2/10/91	24
LSC001	6126	3/10/92	18
LSC001	1214	4/10/92	18
LSC001	8191	1/11/92	12
PAG02	8191	1/05/93	12
PAG02	4112	4/01/91	24
PAG02	6126	1/11/92	12

Tabelas na 2FN

Emp

CodEmp	Nome	Cat	Sal
2146	João	A1	4
3145	Sílvio	A2	4
1214	Carlos	A2	4
8191	Mário	A1	4
4112	João	A2	4
6126	José M.	B1	9

TERCEIRA FORMA NORMAL

Terceira forma normal - 3FN

- Onde está a redundância aqui?

Emp

CodEmp	Nome	Cat	Sal
2146	João	A1	4
3145	Sílvio	A2	4
1214	Carlos	A2	4
8191	Mário	A1	4
4112	João	A2	4
6126	José M.	B1	9

Terceira forma normal - 3FN

- Onde está a redundância aqui?

Emp

CodEmp	Nome	Cat	Sal
2146	João	A1	4
3145	Sílvio	A2	4
1214	Carlos	A2	4
8191	Mário	A1	4
4112	João	A2	4
6126	José M.	B1	9

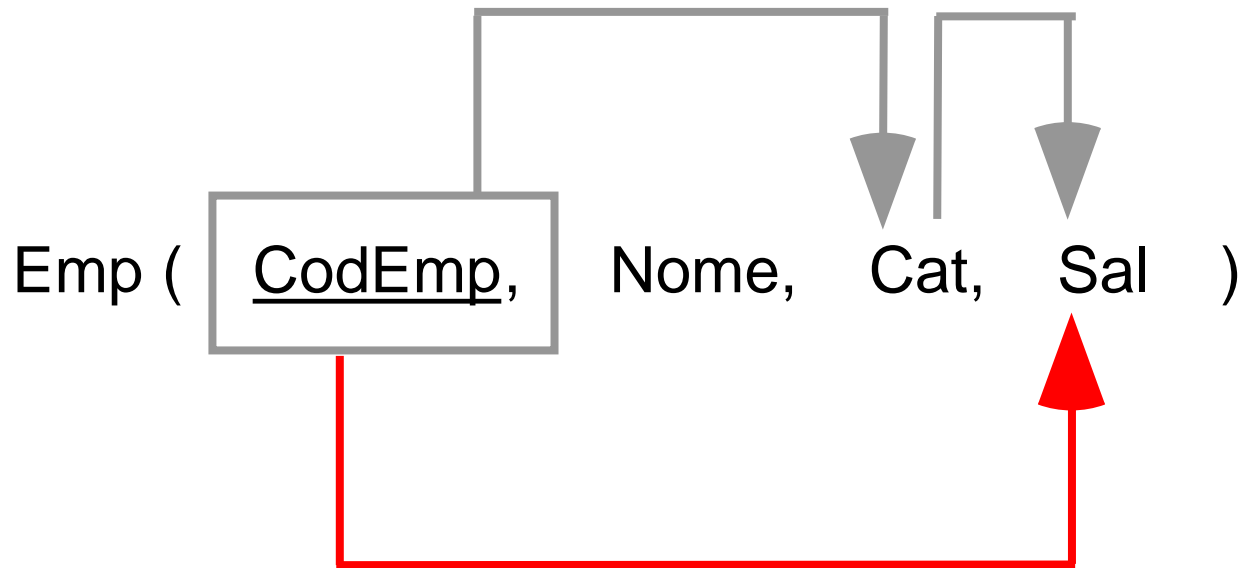
Terceira forma normal - 3FN

- Exemplo de situação indesejada
 - A troca do salário de uma categoria precisa ser feita em vários lugares

Emp

CodEmp	Nome	Cat	Sal
2146	João	A1	5
3145	Sílvio	A2	4
1214	Carlos	A2	4
8191	Mário	A1	5
4112	João	A2	4
6126	José M.	B1	9

Dependência transitiva



Dependência funcional transitiva (indireta)

- Dependências funcionais
 - `codEmp` \rightarrow `nome`, `cat`, `sal`
 - `cat` \rightarrow `sal`

Terceira forma normal - 3FN

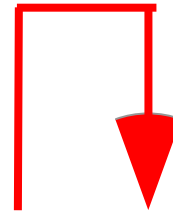
terceira forma normal (3FN)

=

uma tabela encontra-se na terceira forma normal, quando, além de estar na 2FN, não contém dependências transitivas

Dependência transitiva

2FN



Emp (CodEmp, Nome, Cat, Sal)

3FN

Emp (CodEmp, Nome, Cat)

Cat referencia Categ

Categ (Cat, Sal)

Normalização do exemplo

ÑN

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

1FN

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

2FN

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

3FN

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

Tabelas na 3FN

Proj

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

ProjEmp

CódProj	CodEmp	DataIni	TempAI
LSC001	2146	1/11/91	24
LSC001	3145	2/10/91	24
LSC001	6126	3/10/92	18
LSC001	1214	4/10/92	18
LSC001	8191	1/11/92	12
PAG02	8191	1/05/93	12
PAG02	4112	4/01/91	24
PAG02	6126	1/11/92	12

Tabelas na 3FN

Emp

CodEmp	Nome	Cat
2146	João	A1
3145	Sílvio	A2
1214	Carlos	A2
8191	Mário	A1
4112	João	A2
6126	José M.	B1

Categ

Cat	Sal
A1	5
A2	4
B1	9

Exercício

- Os slides a seguir mostram a estrutura utilizada em códigos fontes para representar dados de alunos.
- Com base nesses arquivos, gere um modelo de dados que satisfaça a terceira forma normal.

Exercício - Código fonte PASCAL

```
type reg_aluno= record
  cod_al: integer;
  nome_al: char_60;
  ingressos_cursos_al: array [1..10] of record
    cod_curso: integer;
    semestre_ingresso: integer
  end;
  disciplinas_cursadas_al: array [0..200] of record
    cod_disc: integer;
    semestres_cursados: array [1..20] of record
      semestre_disc: integer;
      nota_disc: integer
    end
  end
end;
arq_aluno= file of reg_aluno;
```

Exercício - Código fonte COBOL

```
FD    Arq-Alunos
01    Reg-Al.
      03    Cod-Al
      03    Nome-Al
      03    Ingr-Cursos-al OCCURS 1 TO 10
            05    Cod-Curso
            05    Sem-ingresso
      03    Disc-Curs-Al OCCURS 0 to 200
            05    Cod-Disc
            05    Sem-Cursado OCCURS 1 TO 20
                  07    Sem-Disc-Cursada
                  07    Nota-Disc
```


Exercício - Esquema ÑN correspondente

Arq-Alunos (Cod-Al, Nome-Al,
(Cod-Curso, Sem-ingresso),
(Cod-Disc,
(Sem-Disc-Cursada, Nota-Disc)))

Atividade Individual

- Passe o modelo abaixo para a terceira forma normal

Data	Jogador	Cartão	Tempo	Time	Juiz	País
15/06/2014	Higuaita	amarelo	23:00	Colômbia	Meira Ricci	Brasil
15/06/2014	Maradona	vermelho	11:00	Argentina	Óscar Ruiz	Colômbia
16/06/2014	Messi	amarelo	60:00	Argentina	Amarilla	Paraguai
16/06/2014	F. Melo	amarelo	14:00	Brasil	Óscar Ruiz	Colômbia
16/06/2014	F. Melo	vermelho	24:00	Brasil	Óscar Ruiz	Colômbia
16/06/2014	Messi	amarelo	23:00	Argentina	Amarilla	Paraguai