

Banco de Dados

Normalização de Dados

Professor: Paulo Sérgio Ruiz Del Aguila

Normalização

- Conjunto de regras aplicadas numa tabela com o objetivo de corrigir possíveis erros de projeto, de modo a obter o máximo de independência de dados, eliminando redundâncias desnecessárias.
- Tal procedimento é feito a partir da identificação de uma anomalia em uma relação, decompondo-as em relações melhor estruturadas.

Normalização

- Objetivos para aplicar a normalização em tabelas:
 - Eliminar redundância de dados.
 - Eliminar estruturas do MER inexistentes no modelo lógico.
- As regras são chamadas Formas Normais:
 - 1ª Forma Normal (1FN).
 - 2ª Forma Normal (2FN).
 - 3ª Forma Normal (3FN).

3

Normalização



4

Anomalias

- Anomalia de Inclusão
- Anomalia de Exclusão
- Anomalia de Alteração

5

Anomalias

- Anomalia de Inclusão

Pedido							
NumPedido	prazo	cliente	endereco	cidade	uf	inscEstadual	codProd
unidade	quantidade	descricao	valorUnit	totalProd	totalPedido	codVendedor	nomeVendedor

Pedido (numPedido, prazo, cliente, endereco, cidade, uf, inscEstadual, codProd, unidade, quantidade, descricao, valorUnit, totalProd, totalPedido, codVendedor, nomeVendedor)

- Um novo cliente só poderá ser incluído se estiver relacionado a uma venda.

6

Anomalias

- Anomalia de Exclusão

Pedido							
NumPedido	prazo	cliente	endereco	cidade	uf	inscEstadual	codProd
unidade	quantidade	descricao	valorUnit	totalProd	totalPedido	codVendedor	nomeVendedor

Pedido (numPedido, prazo, cliente, endereço, cidade, uf, inscEstadual, codProd, unidade, quantidade, descricao, valorUnit, totalProd, totalPedido, codVendedor, nomeVendedor)

- Ao ser excluído um cliente, os dados referentes às suas compras serão perdidos.

7

Anomalias

- Anomalia de Alteração

Pedido							
NumPedido	prazo	cliente	endereco	cidade	uf	inscEstadual	codProd
unidade	quantidade	descricao	valorUnit	totalProd	totalPedido	codVendedor	nomeVendedor

Pedido (numPedido, prazo, cliente, endereço, cidade, uf, inscEstadual, codProd, unidade, quantidade, descricao, valorUnit, totalProd, totalPedido, codVendedor, nomeVendedor)

- Caso se altere o preço de um produto, será preciso percorrer toda a relação para se realizar múltiplas alterações.

8

Normalização

- Permite identificar a existência de problemas potenciais (anomalias de atualização) no projeto de um BD relacional.
- Consiste em gradativamente retirar das relações do esquema as dependências funcionais indesejáveis. Cada um dos passos do processo coloca a relação em uma das formas normais.

9

Normalização

- Sendo bem sucedida:
 - O espaço de armazenamento dos dados diminui.
 - A tabela pode ser atualizada com maior eficiência.

10

Dependência Funcional

- Principal ferramenta de avaliação para identificar se o agrupamento de atributos de uma tabela é apropriado.
- Resolvendo a dependência, procuramos evitar:
 - Redundância de dados.
 - Inconsistências.
 - Perda de dados em operações de remoções ou alterações.

11

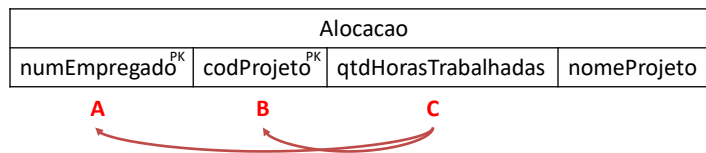
Dependência Funcional

- Existem 3 tipos:
 - Dependência Funcional Total.
 - Dependência Funcional Parcial.
 - Dependência Funcional Transitiva.

12

Dependência Funcional Total

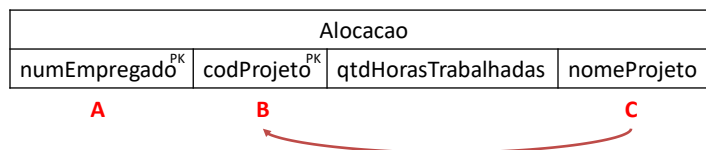
- Os atributos não chave de uma tabela têm que depender da chave primária e somente dela.
- Ex. Uma determinada tabela possui sua chave primária composta pelos atributos A e B. Logo, C será dependente funcional total se e somente se C depender funcionalmente de A e B.



13

Dependência Funcional Parcial

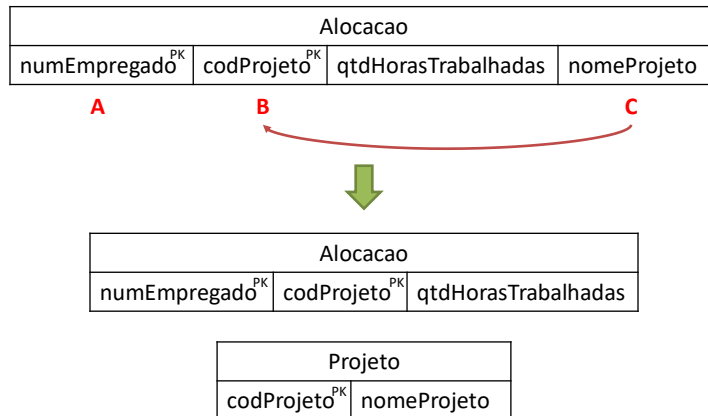
- Os atributos não chave de uma tabela dependem de parte da chave primária.
- Ex. Uma determinada tabela possui sua chave primária composta pelos atributos A e B. Logo, C será dependente funcional parcial se e somente se C depender funcionalmente de A **ou** B.



14

Dependência Funcional Parcial

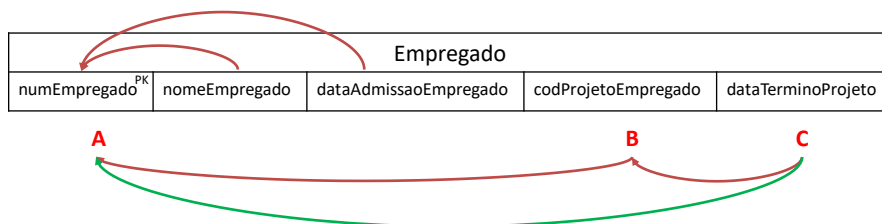
- Solução para a dependência funcional parcial



15

Dependência Funcional Transitiva

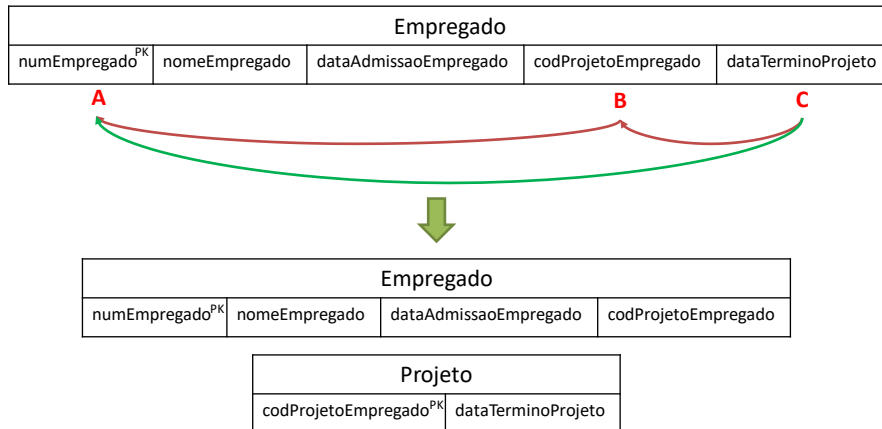
- Os atributo C é dependente funcional transitivo de A se C é funcionalmente dependente de B e B é funcionalmente dependente de A, na mesma tabela.



16

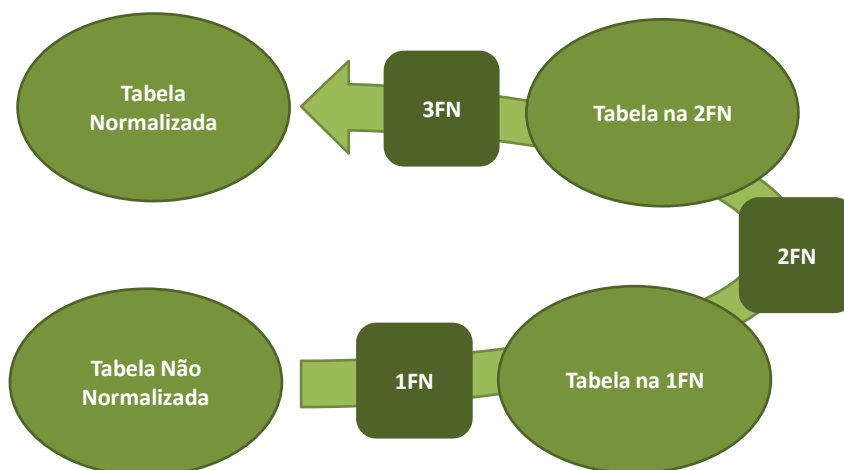
Dependência Funcional Transitiva

- Solução para a dependência funcional transitiva



17

Normalização



18

1ª Forma Normal (1FN)

- Dizemos que uma tabela está na primeira forma normal, quando ela não possui:
 - Atributos multivalorados.
 - Atributos compostos.
 - Obs. Atributos também conhecidos como tabelas aninhadas.

19

1ª Forma Normal (1FN)

- Uma tabela está na 1FN se todos os atributos que a compõem são atômicos, ou seja, se todas as colunas que a compõem são atômicas e monovaloradas.
- Não admite repetições ou colunas que tenham mais que um valor.

20

1ª Forma Normal (1FN)

- Quais os problemas em ter esse tipo de atributo?

Cliente			
codigo	nome	endereço	telefone
1001	Maria do Carmo	Rua Socorro de Medeiros 256 Bairro Cidade Nova	987456523 996587425
1002	Antonio Silva	Rua Comandante Lopes 746 Bairro Adrianopolis	981657423 997546321 992546978

21

1ª Forma Normal (1FN)

- Problemas observados:
 - Temos clientes com mais de um número de telefone.
 - Temos clientes com endereço dividido em diversos valores como rua, número e bairro.

22

1ª Forma Normal (1FN)

- Resolvendo o atributo composto:

Cliente				
codigo	nome	rua	numero	bairro
1001	Maria do Carmo	Rua Socorro de Medeiros	256	Cidade Nova
1002	Antonio Silva	Rua Comandante Lopes	746	Adrianopolis

23

1ª Forma Normal (1FN)

- Resolvendo o atributo multivalorado:

TelefoneCliente	
codigoCli	Telefone
1001	987456523
1001	996587425
1002	981657423
1002	997546321
1002	992546978

24

1ª Forma Normal (1FN)

- Tabelas na 1FN:

Cliente				
codigo	nome	rua	numero	bairro
1001	Maria do Carmo	Rua Socorro de Medeiros	256	Cidade Nova
1002	Antonio Silva	Rua Comandante Lopes	746	Adrianopolis

TelefoneCliente	
codigoCli	Telefone
1001	987456523
1001	996587425
1002	981657423
1002	997546321
1002	992546978

25

2ª Forma Normal (2FN)

- Uma tabela está na 2FN se ela já estiver na 1FN.
- E se todo atributo que não for chave primária for dependente funcional **TOTAL**.

Locacao				
codigoLoc	codFilme	tituloFilme	dataDevolucao	codCliente
1010	201	Parque dos Dinossauros	2020-05-15	123
1011	302	Titanic	2020-05-12	456
1012	201	Parque dos Dinossauros	2020-05-23	789

26

2ª Forma Normal (2FN)

- Problemas observados:
 - A chave primária da tabela Locação são as colunas código da locação e código do filme.
 - A coluna título do filme **não** é dependente funcional **TOTAL**, pois depende somente da coluna código do filme, ou seja de apenas parte da chave primária da tabela Locação.
 - Portanto a coluna título do filme é dependente funcional **PARCIAL** da tabela Locação.

27

2ª Forma Normal (2FN)

- Problemas observados:
 - Se em algum momento tivermos que alterar o título de um filme, teríamos que procurar e alterar os valores em cada linha da tabela.

28

2ª Forma Normal (2FN)

- Resolvendo para a 2FN:
 - Devemos criar uma tabela Filme com as colunas código do filme e título do filme.
 - Devemos vincular por meio de uma chave estrangeira as tabelas Locação e Filme.

29

2ª Forma Normal (2FN)

- Tabelas na 2FN:

Locacao			
codigoLoc	codFilme	dataDevolucao	codCliente
1010	201	2020-05-15	123
1011	302	2020-05-12	456
1012	201	2020-05-23	789

Filme	
codFilme	tituloFilme
201	Parque dos Dinossauros
302	Titanic

30

3ª Forma Normal (3FN)

- Uma tabela está na 3FN se ela já estiver na 2FN.
- E se **NÃO** existir dependência funcional **TRANSITIVA** entre atributos não chave.

Carro				
placa	modelo	kilometragem	codFabricante	nomeFabricante
QWE1234	Polo	8765	3004	Volkswagen
ASD465	Onix	9630	3005	Chevrolet

31

3ª Forma Normal (3FN)

- Problemas observados:
 - A chave primária da tabela Carro é a coluna placa.
 - A coluna nome do fabricante não depende da chave primária da tabela, depende da coluna código do fabricante, que por sua vez, depende da chave primária da tabela.
 - Portanto a coluna nome do fabricante é dependente funcional **TRANSITIVA** da tabela Carro.

32

3ª Forma Normal (3FN)

- Problemas observados:
 - Se em algum momento tivermos que alterar o nome de um fabricante, teríamos que procurar e alterar os valores em cada linha da tabela.

33

3ª Forma Normal (3FN)

- Resolvendo para a 3FN:
 - Devemos criar uma tabela Fabricante com as colunas código do fabricante e nome do fabricante.
 - Devemos vincular por meio de uma chave estrangeira as tabelas Carro e Fabricante.

34

3ª Forma Normal (3FN)

- Tabelas na 3FN:

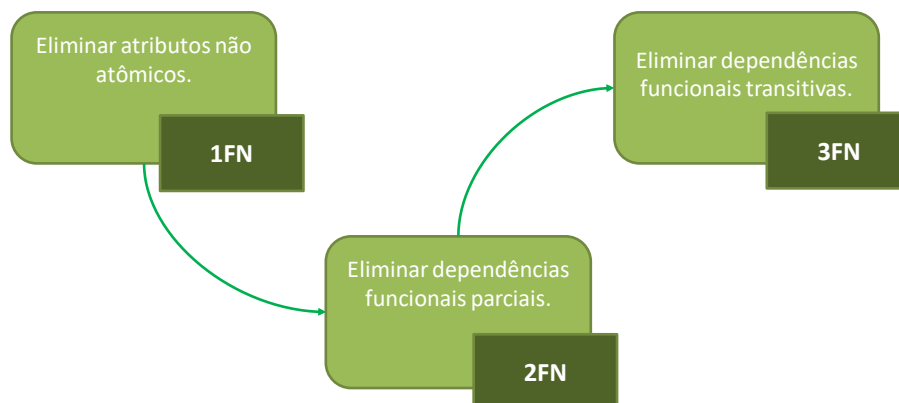
Carro			
placa	modelo	kilometragem	codFabricante
QWE1234	Polo	8765	3004
ASD465	Onix	9630	3005

Fabricante	
codFabricante	nomeFabricante
3004	Volkswagen
3005	Chevrolet

35

Normalização

- Resumo:



36

Exercício

- Normalize a tabela abaixo:

IFAM – Campus Manaus Zona Leste				
Curso: Engenharia de Software Código do curso: 0020				
Aluno: Mateus Bentes Matrícula: 201945 Status: Cursando				
Histórico				
Disciplina (código)	Professor - código	Nota	Faltas	Situação
Banco de Dados (ESW015)	Paulo – 001	8,5	4	Aprovado
Algoritmo e Estrutura de Dados 2 (ESW013)	Benevaldo – 002	9,0	3	Aprovado
Probabilidade e Estatística (ESW016)	Andreia – 003	5,5	8	Reprovado

37

Exercício

- Normalizando a tabela (1FN)
 - Eliminamos os atributos compostos.

Aluno						
matriculaAluno	nomeAluno	statusAluno	unidade	codigoCurso	nomeCurso	
201945	Mateus Bentes	Cursando	CMZL	0020	Engenharia de Software	
Historico						
codigoDisciplina	nomeDisciplina	codigoProfessor	nomeProfessor	nota	numeroFaltas	situacao
ESW015	Banco de Dados	001	Paulo	8,5	4	Aprovado
ESW13	Algoritmo e Estrutura de Dados 2	002	Benevaldo	9,0	3	Aprovado
ESW16	Probabilidade e Estatística	003	Andreia	5,5	8	Reprovado

38

Exercício

- Normalizando a tabela (1FN)
 - Histórico é uma tabela aninhada (atributo multivalorado), então temos que dividir, mas sem perder a referência.

Aluno					
matriculaAluno	nomeAluno	statusAluno	unidade	codigoCurso	nomeCurso
201945	Mateus Bentes	Cursando	CMZL	0020	Engenharia de Software

Historico							
matriculaAluno	codigoDisciplina	nomeDisciplina	codigoProfessor	nomeProfessor	nota	numeroFaltas	situacao
201945	ESW015	Banco de Dados	001	Paulo	8,5	4	Aprovado
201945	ESW13	Algoritmo e Estrutura de Dados 2	002	Benevaldo	9,0	3	Aprovado
201945	ESW16	Probabilidade e Estatística	003	Andreia	5,5	8	Reprovado

39

Exercício

- Normalizando a tabela (2FN)
 - A tabela Aluno tem como PK somente campo matricula do aluno, então não temos casos de dependência funcional parcial.
 - Portanto a tabela Aluno já se encontra na 2FN.

Aluno					
matriculaAluno	nomeAluno	statusAluno	unidade	codigoCurso	nomeCurso
201945	Mateus Bentes	Cursando	CMZL	0020	Engenharia de Software

40

Exercício

- Normalizando a tabela (2FN)
 - Na tabela Histórico temos uma PK composta pelos campos matrícula do aluno, código da disciplina, e código do professor.
 - Então temos dependência transitiva parcial com os campos nome da disciplina e nome do professor.

Historico							
matriculaAluno	codigoDisciplina	nomeDisciplina	codigoProfessor	nomeProfessor	nota	numeroFaltas	situacao
201945	ESW015	Banco de Dados	001	Paulo	8,5	4	Aprovado
201945	ESW13	Algoritmo e Estrutura de Dados 2	002	Benevaldo	9,0	3	Aprovado
201945	ESW16	Probabilidade e Estatística	003	Andreia	5,5	8	Reprovado

41

Exercício

- Normalizando a tabela (2FN)

Historico					
matriculaAluno	codigoDisciplina	codigoProfessor	nota	numeroFaltas	situacao
201945	ESW015	001	8,5	4	Aprovado
201945	ESW13	002	9,0	3	Aprovado
201945	ESW16	003	5,5	8	Reprovado

Disciplina	
codigoDisciplina	nomeDisciplina
ESW015	Banco de Dados
ESW13	Algoritmo e Estrutura de Dados 2
ESW16	Probabilidade e Estatística

Professor	
codigoProfessor	nomeProfessor
001	Paulo
002	Benevaldo
003	Andreia

42

Exercício

- Normalizando a tabela (3FN)
 - Na tabela Aluno temos o campo nome do curso que tem dependência funcional transitiva com o campo matrícula do aluno que forma a PK da tabela Aluno.

Aluno					
matriculaAluno	nomeAluno	statusAluno	unidade	codigoCurso	nomeCurso
201945	Mateus Bentes	Cursando	CMZL	0020	Engenharia de Software

43

Exercício

- Normalizando a tabela (3FN)

Aluno				
matriculaAluno	nomeAluno	statusAluno	unidade	codigoCurso
201945	Mateus Bentes	Cursando	CMZL	0020

Curso	
codigoCurso	nomeCurso
0020	Engenharia de Software

44

Exercício

- Normalizando a tabela (3FN)
 - As tabelas Histórico, Disciplina e Professor já se encontram na 3FN.

Historico					
matriculaAluno	codigoDisciplina	codigoProfessor	nota	numeroFaltas	situacao
201945	ESW015	001	8,5	4	Aprovado
201945	ESW13	002	9,0	3	Aprovado
201945	ESW16	003	5,5	8	Reprovado

Disciplina	
codigoDisciplina	nomeDisciplina
ESW015	Banco de Dados
ESW13	Algoritmo e Estrutura de Dados 2
ESW16	Probabilidade e Estatística

Professor	
codigoProfessor	nomeProfessor
001	Paulo
002	Benevaldo
003	Andreia

45

Exercício

- Tabelas normalizadas

Aluno				
matriculaAluno	nomeAluno	statusAluno	unidade	codigoCurso
201945	Mateus Bentes	Cursando	CMZL	0020

Curso	
codigoCurso	nomeCurso
0020	Engenharia de Software

Professor	
codigoProfessor	nomeProfessor
001	Paulo
002	Benevaldo
003	Andreia

Historico					
matriculaAluno	codigoDisciplina	codigoProfessor	nota	numeroFaltas	situacao
201945	ESW015	001	8,5	4	Aprovado
201945	ESW13	002	9,0	3	Aprovado
201945	ESW16	003	5,5	8	Reprovado

Disciplina	
codigoDisciplina	nomeDisciplina
ESW015	Banco de Dados
ESW13	Algoritmo e Estrutura de Dados 2
ESW16	Probabilidade e Estatística

46

Obrigado

E-mail: paulo.aguila@ifam.edu.br

Telefone: (92) 98189-8899