

53 45 52 45 49 20 46 49 45 4c 20
41 4f 53 20 50 52 45 43 45 49 54
4f 53 20 44 41 20 48 4f 4e 52 41
20 45 20 44 41 20 43 49 c3 8a 4e
43 49 41 2c 20 50 52 4f 4d 4f 56
45 4e 44 4f 20 4f 20 55 53 4f 20
45 20 4f 20 44 45 53 45 4e 56 4f
4c 56 49 4d 45 4e 54 4f 20 44 41
20 49 4e 46 4f 52 4d c3 81 54 49
43 41 20 45 4d 20 42 45 4e 45 46
c3 8d 43 49 4f 20 44 4f 20 43 49
44 41 44 c3 83 4f 20 45 20 44 41
20 53 4f 43 49 45 44 41 44 45 2e

RESIDÊNCIA DE SOFTWARE

**CAPACITAR
TREINAR
EMPREGAR**

TRANSFORMAR



Banco de Dados



Aula 3
Normalização

Roni Schanuel
01-04-2022

Dicionário de Dados

Um dicionário de dados é um documeto usado para descrever informações sobre o conteúdo, formato e a estrutura do banco de dados, junto com o modelo de entidade e relacionamento, é necessário que se mantenha um documento com a explicação de todos os objetos nele criados. Este documento, que pode ser chamado de dicionário de dados, permite que os analistas obtenham informações sobre todos os objetos do modelo de forma textual, contendo explicações por vezes difíceis de incluir no diagrama. O dicionário de dados é geralmente criado em forma de tabelas.

| Tabela: Funcionário | | | | | |
|---------------------|--------------|----------|---------------|--------------|-----------------------------------------|
| Atributo | Tipo/Domínio | Tamanho | Restrições | Valor Padrão | Descrição |
| codigo_funcionario | Numérico | 4 bytes | PK - NOT NULL | N/D | Número gerado automaticamente |
| nome | Texto | 40 bytes | NOT NULL | N/D | Nome do funcionário da empresa |
| id_departamento | Numérico | 4 bytes | FK | N/D | Número de identificação do departamento |

Normalização

É um processo que aplica um conjunto de regras sobre o modelo do banco de dados relacional. As tabelas que atendem a um determinado conjunto de regras estão em uma determinada forma normal. Algumas características são:

- ❖ Verificar se o modelo está corretamente projetado
- ❖ Permitir o armazenamento consistente
- ❖ Permitir eficiente acesso aos dados

Evita problemas como:

- ❖ Inconsistência nos dados
- ❖ Redundância
- ❖ Falta de integridade

Normalização

A utilização da normalização é necessária porque podem ocorrer erros de modelagem, utilizamos a técnica para garantir a integridade de um modelo de dados. Através do processo de normalização pode-se, gradativamente, substituir um conjunto de entidades e relacionamentos por um outro, mais eficiente em relação às anomalias de atualizações

Primeira Forma Normal

Uma tabela está na 1FN significa que os valores dos atributos são atômicos não podem se repetir e também não podem possuir atributos multivalorados com mais de um valor.

Primeira Forma Normal

Uma tabela está na 1FN significa que os valores dos atributos são atômicos não podem se repetir e também não podem possui atributos multivalorados com mais de um valor.

| Funcionario | codigo | nome | email | telefone | endereco |
|-------------|--------|--------|---------------------|------------------------------|-----------------------------------------------|
| codigo | 100 | Manoel | manoel@gmail.com | 24-2334-1989 24-2234-1990 | Rua do Imperador, 285, Centro, Petrópolis, RJ |
| nome | 120 | Ana | ana@hotmail.com | 24-2238-1299 | Rua Bingen, 110, Bingen, Petrópolis, RJ |
| email | 121 | Marcos | marcos@yahoo.com.br | 24-2221-0909 24-2223-1090 | Rua Fonseca Ramos 185, Centro, Petrópolis, RJ |
| telefone | | | | | |
| endereco | | | | | |

Passagem para 1FN

Encontrar a chave primária da tabela, procurar as colunas da tabela que apresentam dados repetidos para que sejam removidas, crie uma tabela para esses dados repetidos, por fim, estabeleça relação entre a nova tabela e a principal.

| Funcionario | | | | | | |
|-------------|--------|---------------------|--------------------------------|--------|------------|--------|
| codigo | nome | email | rua | bairro | cidade | estado |
| 100 | Manoel | manoel@gmail.com | Rua do Imperador, 285 | Centro | Petrópolis | RJ |
| 120 | Ana | ana@hotmail.com | Rua Bingen, 110 Petrópolis, RJ | Bingen | Petrópolis | RJ |
| 121 | Marcos | marcos@yahoo.com.br | Rua Fonseca Ramos 185 | Centro | Petrópolis | RJ |

| Telefone | |
|--------------------|--------------|
| codigo_funcionario | telefone |
| 100 | 24-2334-1989 |
| 100 | 24-2334-1990 |
| 120 | 24-2238-1299 |
| 121 | 24-2221-0909 |
| 121 | 24-2223-1090 |

Problemas de uma tabela não normalizada na 1FN

A primeira forma normal tenta resolver um dos maiores problemas de banco de dados que é redundância e a desorganização. O campo telefone não pode permitir a entrada de mais de um valor, isto acarretaria em problemas na busca de um dos valores, por exemplo. Outro problema seria um campo endereço onde as partes não estejam desmembradas.

Como seria possível fazer uma busca por endereços de determinado bairro apenas ou de determinadas cidades? Veja que a normalização irá trazer inúmeros benefícios de performance do banco e claro nos possibilitaria trabalhar com esses dados da forma que fosse necessário.

Segunda Forma Normal

Uma relação está na 2FN se, e somente se, estiver na 1FN, cada atributo não chave não poderá ser dependente de apenas parte da chave primária

| Equipamento |
|------------------|
| *cod_equipamento |
| *cod_fornecedor |
| local_fornecedor |
| qtde_estoque |
| email_fornecedor |

| Equipamento | | | | |
|-----------------|----------------|------------------|--------------|------------------|
| cod_equipamento | cod_fornecedor | local_fornecedor | qtde_estoque | email_fornecedor |
| 10 | 135 | Rio de Janeiro | 10 | abc@gmail.com |
| 25 | 138 | São Paulo | 20 | xpto@hotmail.com |
| 54 | 135 | Rio de Janeiro | 25 | xyz@gmail.com |

local do fornecedor e email do fornecedor dependem parcialmente da chave primária composta, sendo assim vamos aplicar a segunda forma normal criando uma tabela separada.

Segunda Forma Normal

Aplicando a segunda forma normal.

| Fornecedor | | |
|----------------|------------------|------------------|
| cod_fornecedor | local_fornecedor | email_fornecedor |
| 135 | Rio de Janeiro | abc@gmail.com |
| 138 | São Paulo | xpto@hotmail.com |

| Equipamento | | |
|-----------------|----------------|--------------|
| cod Equipamento | cod_fornecedor | qtde_estoque |
| 10 | 135 | 10 |
| 25 | 138 | 20 |
| 54 | 135 | 25 |

A segunda forma normal trata destas anomalias e evita que valores fiquem em redundância no banco de dados.

Procedimentos:

- a) Identificar os atributos que não são funcionalmente dependentes de toda a chave primária;
- b) Remover da entidade todos esses atributos identificados e criar uma nova entidade com eles.

Considerando a tabela que não está na 2FN:

| N_pedido | Codigo_produto | Produto | Quant | Valor_unit | Subtotal |
|----------|----------------|----------------------|-------|------------|----------|
| 1005 | 1-934 | Impressora laser | 5 | 1.500,00 | 7.500,00 |
| 1006 | 1-956 | Impressora desjet | 3 | 350,00 | 1.050,00 |
| 1007 | 1-923 | Impressora matricial | 1 | 190,00 | 190,00 |
| 1008 | 1-908 | Impressora mobile | 6 | 980,00 | 5.880,00 |

O nome do produto depende do código do produto, porém não depende de N_pedido que é a chave primária da tabela, portanto não está na segunda forma normal. Isto gera problemas com a manutenção dos dados, pois se houver alteração no nome do produto teremos que alterar em todos os registros da tabela pedido.

Para normalizar esta tabela vamos criar a tabela de Produto

| Codigo_produto | Produto |
|----------------|----------------------|
| 1-934 | Impressora laser |
| 1-956 | Impressora desjet |
| 1-923 | Impressora matricial |
| 1-908 | Impressora mobile |

Agora a tabela de pedido está na 2FN

| N_pedido | Codigo_produto | Quant | Valor_unit | Subtotal |
|----------|----------------|-------|------------|----------|
| 1005 | 1-934 | 5 | 1.500,00 | 7.500,00 |
| 1006 | 1-956 | 3 | 350,00 | 1.050,00 |
| 1007 | 1-923 | 1 | 190,00 | 190,00 |
| 1008 | 1-908 | 6 | 980,00 | 5.880,00 |

Terceira Forma Normal

Assim como para estar na segunda forma normal é preciso estar na primeira forma normal, para estar na terceira forma normal é preciso estar também na segunda forma normal. Toda coluna derivada a partir de outra coluna não chave deve ser retirada do modelo dados. No exemplo abaixo o total depende do preço e quantidade

| Venda | | | | |
|---------------|----------------|------------|--------|--------|
| numero_pedido | codigo_produto | quantidade | valor | total |
| 100 | 3 | 5 | 100,00 | 500,00 |
| 102 | 5 | 3 | 30,00 | 90,00 |
| 106 | 6 | 2 | 10,00 | 20,00 |

Terceira Forma Normal

Em nossa tabela Venda, o subtotal é o resultado da multiplicação de quantidade com valor, desta forma a coluna subtotal depende de outras colunas não-chave, sendo um campo calculado.

Para normalizar esta tabela na terceira forma normal teremos de eliminar a coluna subtotal, como no exemplo a seguir:

| Venda | | | |
|---------------|----------------|------------|--------|
| numero_pedido | codigo_produto | quantidade | valor |
| 100 | 3 | 5 | 100,00 |
| 102 | 5 | 3 | 30,00 |
| 106 | 6 | 2 | 10,00 |

Resumindo

1ª FN

Quando a tabela só possui atributos atômicos (não multivalorados)

2ª FN

Quando estiver na 1FN e todo atributo não-chave for dependente da chave primária

3ª FN

Quando estiver na 2FN e não tiver atributos não-chave dependendo de atributos não-chave

Exercícios

Considere a tabela abaixo e normalize até a terceira forma normal.

| Projeto_Alocacao | Projeto_Alocacao | | | | | | | | |
|----------------------|------------------|--------------|----------------------|--------------------|------------------|--------------|----------------------|---------------|----------------------|
| codigo_projeto | codigo_projeto | tipo_projeto | descricao_projeto | codigo_funcionario | nome_funcionario | codigo_cargo | nome_cargo | salario_cargo | data_inicio_alocacao |
| tipo_projeto | | | | | | | | | |
| descricao_projeto | | | | | | | | | |
| codigo_funcionario | | | | | | | | | |
| nome_funcionario | 10 | Sistema Web | Sistema Ambulatorial | 2 | Marcos | 1 | Analista de Sistemas | 3000,00 | 18/02/2022 |
| codigo_cargo | 25 | Aplicativo | Aplicativo Trânsito | 4 | Joaquim | 2 | Programador | 2500,00 | 18/02/2022 |
| nome_cargo | 25 | Aplicativo | Aplicativo Trânsito | 2 | Marcos | 1 | Analista de Sistemas | 3000,00 | 1802/2022 |
| salario | | | | | | | | | |
| data_inicio_alocacao | | | | | | | | | |

Um projeto pode ter varios funcionários alocados a partir de uma data inicial de alocação. Os funcionários possuem um cargo e são remunerados em função do cargo. Um funcionário pode ser alocado em mais de um projeto.

Resposta Exercício

Projeto

| codigo_projeto | tipo_projeto | descricao_projeto |
|----------------|--------------|----------------------|
| 10 | Sistema Web | Sistema Ambulatorial |
| 25 | Aplicativo | Aplicativo Trânsito |

Cargo

| codigo_cargo | nome_cargo | salario_cargo |
|--------------|----------------------|---------------|
| 1 | Analista de Sistemas | 3000,00 |
| 2 | Programador | 2500,00 |

Alocacao

| codigo_projeto | codigo_funcionario | data_inicio_alocacao |
|----------------|--------------------|----------------------|
| 10 | 2 | 18/02/2022 |
| 25 | 4 | 18/02/2022 |
| 25 | 2 | 1802/2022 |

Funcionario

| codigo_funcionario | nome_funcionario | codigo_cargo |
|--------------------|------------------|--------------|
| 2 | Marcos | 1 |
| 4 | Joaquim | 2 |

Exercícios

Uma faculdade criou um histórico com os dados dos alunos, cursos, professores e turmas. Porém há muitos problemas na atualização deste cadastro pois as informações estão duplicadas e outros campos que podem ser normalizados. Aplique as formas normais na tabela abaixo:

Historico
matricula
nome_aluno
nota1
nota2
media
situacao
codigo_professor
nome_professor
codigo_disciplina
nome_disciplina
codigo_curso
nome_curso
turno
numero_turma
nome_turma

Resolução Exercício

Historico

matricula
codigo_disciplina
nota1
nota2
situacao

Aluno

matricula
nome
codigo_curso

Curso

codigo_curso
nome
turno

Disciplina

codigo_disciplina
nome_disciplina
codigo_professor

Professor

codigo_professor
nome_professor