Dados x Informação

Dados: São fatos em uma forma primaria, que podem ser armazenados em algum meio.

Ex: CPF, Nome, Data..

Informação: São fatos organizados de maneira a produzir um significado. -> Dados colocados em contexto.

Ex: Lista de Clientes com seus números de CPF, ordenados.

Metadados

Definido como sendo “Dados sobre os Dados”

Permitem efetuar a representação e identificação dos dados, garantindo sua consistência e persistência.

Ficam no Dicionário de Dados

Banco de Dados

Conjunto organizado de Dados. Organizados de modo a modelar aspectos do mundo real, para que seja possível efetuar processamentos que gere informações relevantes para os usuários a partir desses dados.

Composto por diversos objetos como tabelas, esquemas, visões, consultas, relatórios etc.

Podemos aplicar em sistemas bancários, reserva de hotéis, controle de estoque, e-commerce, Receita Federal etc.

Modelagem de Dados

Metodologia utilizada para determinar as Regras de Negocio e a arquitetura de um Banco de Dados, ou seja, descreve as estruturas lógicas e físicas do banco.

SGBD – Sistema de Gerenciamento de Banco de Dados

Coleção de Software que permite aos usuários criarem e manterem um ou mais banco de dados.

Usados para definir, construir, manipular e compartilhar banco de dados entre aplicações e usuários. Permitindo também proteger o BD ao longo do tempo.

SGBD mais conhecidos: MySQL, Microsoft SQL Server, IBM DB2, MongoDB, TeraData, PostgreSQL, SQLite, Oracle DataBase, entre outros.

Diagrama

Descrição gerada automaticamente

(imagem 1: retirada através da função PrintScreen no YouTube dia 01/03/2023)

Usuários

Podendo ser dividido em: Administrator (DBA), Projetista/Desenvolvedor e Usuário Final.

Modelo de Banco de Dados

Modelo Hierárquico

Modelo são organizados de forma hierárquica, com conjuntos de tipos de registros interconectados por meio de ligações. As ligações representam uma relação entre dois tipos de registros: Pai e filho.

Este modelo também chamado de Diagrama de Estrutura em Árvore. As ações de acessos são unidirecionais, a partir do pai ao filho.

Diagrama

Descrição gerada automaticamente

(imagem 2: retirada através da função PrintScreen no YouTube dia 01/03/2023)

Modelo em Rede

Neste modelo em Redes os dados são organizados em tipos e ligações entre dois registros. Não existindo restrição hierárquica.

Tanto o esquema quanto ocorrências de dados, são visualizados como um grafo direcionado.

Interface gráfica do usuário

Descrição gerada automaticamente com confiança média

(imagem 3: retirada através da função PrintScreen no YouTube dia 01/03/2023)

Modelo Relacional

Neste modelo os dados são separados em entidades, conforme cada assunto, e registrados como atributos dessas entidades.

As entidades se relacionam entre si e permitem que os dados sejam armazenados e recuperados de forma rápida e segura.

Diagrama

Descrição gerada automaticamente

(imagem 4: retirada através da função PrintScreen no YouTube dia 01/03/2023)

Modelo

Estrutura que auxilia a comunicar os conceitos que estão na mente do projetista. Podemos usá-los para tarefas como descrever, analisar, especificar e comunicar ideias.

O modelo deve possuir detalhes suficientes para que um desenvolver consiga construir o banco de dados de acordo com a necessidade do projeto.

Modelagem de Dados

Modelo de Dados é o processo de criação de um Modelo de Dados para um sistema de informação, com a aplicação de técnicas especificas de modelagem.

Trata-se de processos para definir e analisar requisitos de dados necessários da regra de negócio para suportar processos com sistemas informatizados em organizações.

Tal modelo fornece uma estrutura para os dados usados em um Sistema de Informação, com definições e formatos específicos.

Modelo Relacional

Neste modelo são organizados em coleções de tabelas bidimensionais. Essas tabelas são chamadas de “Relações”.

Relação é uma forma de se organizar os dados em linhas e colunas baseado em lógica e teoria de conjuntos.

Diagrama

Descrição gerada automaticamente

(imagem 5: retirada através da função PrintScreen no Google Imagens dia 01/03/2023)

Banco de Dados Relacional é uma coleção de relações, que são tabelas bidimensionais, onde os dados são armazenados.

Componentes de um Banco de Dados

Tabela: Estrutura básica de armazenamento no SGBDR (Sistema de Gerenciamento do Banco de Dados Relacional). Armazena todos os dados necessários sobre algo do mundo real, como clientes, pedidos ou produtos

Tupla: ou linha/registro, representa todos os dados requeridos por uma determinada ocorrência de entidade em particular. Por exemplo, os dados de um cliente específico.

Cada linha em uma tabela deve ser identificada por uma chave primária, de modo a não haver duplicação de registros.

Coluna: Unidade que armazena um tipo específico de dado (valor) – ou não armazena nada, com valor nulo.

Tela de jogo de vídeo game

Descrição gerada automaticamente

(imagem 6: retirada através da função PrintScreen no Google Imagens dia 01/03/2023)

Relacionamento: Associação entre as entidades (tabelas), conectadas por chaves primárias e chaves estrangeiras.

Chave Primária ou Primary Key (PK): Coluna (atributo) que identifica um registro de forma exclusiva na tabela.

Chave Estrangeira ou Foreign Key (FK): Coluna que define como as tabelas se relacionam umas com as outras. Uma FK se refere a uma PK ou a uma Chave Única em outra tabela (ou na mesma tabela).

Tela de jogo de vídeo game

Descrição gerada automaticamente com confiança média

(imagem 7: retirada através da função PrintScreen no Google Imagens dia 01/03/2023)

Análise de Requisitos

São realizadas reuniões para coleta de informações, que analisam o que é exigido para o banco de dados a ser criado. Os processos de negócio são definidos, e as entidades atributos e relacionamentos do Banco de Dados são documentadas.

A Análise é extremamente importante para o sucesso do projeto do Banco de Dados. Nesta etapa é possível levantar quais dados serão ou não armazenados no banco.

Modelo Entidade-Relacionamento

MER, cria um diagrama entidade-relacionamento a partir das especificações do negócio ou narrativas do usuário. Permite ilustrar as entidades em um negócio e relacionamento entre elas. Construímos o MER durante a fase de análise no ciclo de vida de desenvolvimento do sistema.

Um MER separa a informação necessária ao negócio das atividades que são realizadas no negócio. Servindo para ilustrar as entidades do negócio assim como os relacionamentos entre si.

Diagrama

Descrição gerada automaticamente

(imagem 8: retirada através da função PrintScreen no site da Alura dia 01/03/2023)

Logo, a MER, descreve os objetos do mundo real através de entidades, com suas propriedades que são os atributos e os seus relacionamentos.

Diagrama

Descrição gerada automaticamente

(imagem 9: retirada através da função PrintScreen no Google Imagens dia 01/03/2023)

O processo é modelado como componentes (entidades) que são ligadas umas as outras por relacionamentos que indicam as dependências entre elas. As entidades podem ter várias propriedades (atributos) que as caracterizam.

Diagramas são criados para representar graficamente entidades, atributos e relacionamentos, denominados de Diagramas Entidade-Relacionamento (DER)

Diagrama Entidade-Relacionamento (DER)

Trata-se de uma representação gráfica da MER ou parte dele, o modelo traz entidades, atributos e relacionamentos, que constam informações sobre os tipos de dados, restrições, descrições de entidades e outras.

Seus componentes:

Uma imagem contendo Ícone

Descrição gerada automaticamente

(imagem 10: retirada através da função PrintScreen Youtube dia 02/03/2023)

Diagrama

Descrição gerada automaticamente

(imagem 11: retirada através da função PrintScreen Google Imagens dia 02/03/2023)

Entidades: Representam um objeto do mundo real e que possuem uma existência independente, como: pessoas, empresa, carro, casa, entre outras coisas.

Algo de importância para um usuário ou organização que precisa ser representado em um banco de dados. Representa um tema, tópico ou conceito de negócio.

Uma entidade pode ter existência física ou abstrata. São exemplos: Empregados, Livros, Vendas, Produtos etc.

Nomeamos as entidades usando substantivos que representam de forma clara e objetiva sua função. São exemplos: *Produto, Cliente, Venda, Estoque, Catálogo*, entre outras.

São boas práticas:

1. Devem começar com uma letra:

Usar palavras no singular;

Não podem ter espaços ou alguns caracteres especiais;

Alguns caracteres como “$”, “#” e “\_” são permitidos em alguns bancos de dados;

1. Os nomes de colunas devem ser únicos dentro de uma tabela.
2. Os nomes de entidades/tabelas devem ser únicos dentro do esquema.

Instância de Entidade

Descrição da estrutura e formato das ocorrências da entidade. Trata-se de todas as informações que não preenchidas pelo usuário referente uma entidade, cada preenchimento diverge um do outro, chamando de Instância.

As entidades podem ser divididas em fortes, fracas e associativas.

1. Forte: Não depende de outra entidade para existir
2. Fraca: Depende de outra entidade para existir, ou seja, não possuem existência própria ou não possuem atributos próprios para identificação, dependendo dos atributos chave de entidades fortes.
3. Associativas: São utilizadas quando existe a necessidade de associar uma entidade a um relacionamento. Normalmente a relação é muitos para muitos.

Interface gráfica do usuário, Texto, Aplicativo

Descrição gerada automaticamente

(imagem 12: retirada através da função PrintScreen do MySQL Workbench dia 01/03/2023)

Atributos

Descrevem propriedades das entidades. A entidade *pessoa* pode ter como atributo o *nome*, *data de nascimento*, *idade*, *endereço*, o item pode ser obrigatório ou opcional seu preenchimento, através do NULL e NOTNULL.

Podendo ser divididas em: simples, compostos, multivalorados, derivado e chave.

1. Simples: são indivisíveis, ou seja, atributos atômicos, um exemplo seria o atributo *CPF*, dado que não pode ser dividido em partes menores para formar atributos.
2. Composto: podem ser divididos em pares menores, que representam outros atributos, como atributo endereço que pode ser dividido em atributos menores exemplo, *cidade*, *estado*, *rua*, *CEP*.
3. Multivalorado: pode ter um ou vários valores associados a ele, exemplo, atributo *telefone* de um cliente, pode ter um ou vários telefones
4. Derivado: dependem de outro atributo ou até mesmo outra entidade para existir, exemplo, atributo *idade* e o atributo *data de nascimento*.
5. Chave: utilizado para identificar de forma única uma entidade, ou seja, os valores associados a esse atributo são distintos dentro o conjunto de entidades, exemplo, atributo *CPF* de uma pessoa, ele é único e pode ser utilizado como atributo chave.

As chaves podem ser únicas ou não-únicas:

Únicas: O valor dos dados da chave é único na entidade

Não única: Usada para agrupar instâncias de classe em categorias

As chaves podem ser compostas, consistindo em dois ou mais atributos combinados

Tela de jogo de vídeo game

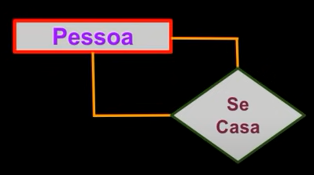
Descrição gerada automaticamente com confiança média

(imagem 13: retirada através da função PrintScreen do Youtube dia 02/03/2023)

Grau de um Relacionamento

Define o número de entidades que participam do relacionamento. Podendo ser:

* Unário



(imagem 13: retirada através da função PrintScreen do Youtube dia 06/03/2023)

* Binário

Uma imagem contendo objeto, relógio, medidor

Descrição gerada automaticamente

(imagem 13: retirada através da função PrintScreen do Youtube dia 06/03/2023)

* Ternário

Diagrama

Descrição gerada automaticamente

(imagem 13: retirada através da função PrintScreen do Youtube dia 06/03/2023)

Relacionamento

Com os dados em diferentes entidades são armazenados em tabelas distintas, geralmente, precisamos combinar duas ou mais tabelas para responder às perguntas específicas dos usuários.

Realizando assim, uma associação nomeada entre as entidades, com um grau de associação, exemplo, *clientes* podem estar associados a *pedidos*.

A relação pode ser bidimensional com características especificas compostas por linhas e colunas, criada a partir de uma entidade.

A ligação seria *“pode ser”* ou *“deve ser”* assim como seu grau de cardinalidade *“um e apenas um”* ou *“um ou mais”.*

Diagrama

Descrição gerada automaticamente

(imagem 14: retirada através da função PrintScreen do MySQL Workbench dia 02/03/2023)

Acima uma demonstração da relação entre a entidade *veículo* com *motorista*, onde a ligação *é um para um*, logo, um veiculo para um motorista.

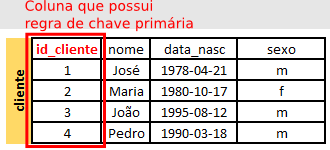
Logo abaixo, a entidade *cliente* com *viagem*, onde a ligação *é um para muitos*, um cliente que pode realizar muitas viagens.

Características de uma relação:

* Linhas contém dados sobre instâncias de uma entidade (registros)
* Colunas contém dados sobre atributos da entidade (campos)
* Cada célula da tabela armazena um único valor
* Todos os valores em uma coluna são do mesmo tipo (domínio)
* Cada coluna possui um nome único
* Não há duas linhas idênticas
* As relações geralmente geram tabelas no banco

Identificador Único (UID)

Qualquer combinação de atributos ou relacionamento que são usados para distinguir ocorrências de uma entidade. Cada ocorrência da entidade deve ser identificável de forma exclusiva.



(imagem 15: retirada através da função PrintScreen do Google Imagens dia 02/03/2023)

Na imagem acima a Entidade *cliente* possui atributos de *id\_cliente*, *nome*, *data\_nasc* e *sexo*, onde o *id\_cliente* possui um Identificador Único, logo sendo uma Chave Primária (PK) ou Chave Estrangeira (FK).

Efetuando relacionamento entre múltiplas tabelas

1. Cada linha de dados em uma tabela deve ser identificada de forma única usando-se uma Chave Primária (UID)
2. Usamos uma Chave Estrangeira para relacionar os dados entre múltiplas tabelas
3. Usamos para isso o relacionamento entre chave primária de uma tabela com a chave estrangeira de outra

Diagrama

Descrição gerada automaticamente

(imagem 15: retirada através da função PrintScreen do Google Imagens dia 06/03/2023)

Chaves

Consiste em uma ou mais colunas e uma relação cujo valores são usados para identificar de formar exclusiva uma linha ou conjunto de linha.

Pode ser única (identifica uma única linha) ou não -única (identifica um conjunto de linhas).

* Únicas (Unique): Candidata, Composta, Primária, Surrogada
* Não-Única (Non-Unique): Estrangeira

Chave Candidata – Atributo ou grupo de atributos com potencial para se tornarem uma chave primária.

Ex: Campos *Num\_Matricula* e *CPF* em uma tabela com registro de alunos.

Identificamos as chaves candidatas que possuem potencial para virar uma chave primária. Caso não seja utilizada como primária ela vira chave alternativa.

Chave Primária PK– Chave candidata escolhida para ser a chave *principal* na relação. Identifica de forma *exclusiva* os registros em uma tabela, não podendo ter repetição de valores nem tampouco valor nulo.

Chave Estrangeira – Coluna de uma tabela que estabelece um relacionamento com a Chave Primária (PK) de outra tabela. É a partir da chave estrangeira (Foreign Key FK) que sabemos com qual registro em outra tabela um registro está relacionado.

Chave Composta – Composta de dois ou mais atributos (colunas). Geralmente empregada quando não é possível utilizar uma única coluna de uma tabela para identificar de forma exclusiva seus registros.

Diagrama

Descrição gerada automaticamente

(imagem 15: retirada através da função PrintScreen do Google Imagens dia 06/03/2023)

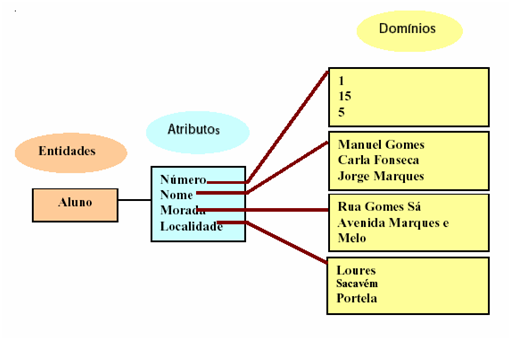
Chave Surrogada/Substituta – Valor número, único, adicionado a uma relação para servir como chave primária. Não possui significado para os usuários e geralmente fica escondida nas aplicações. As chaves substitutas são frequente usadas no lugar de uma chave primária composta.

Instruções para criação de chaves primárias e estrangeiras

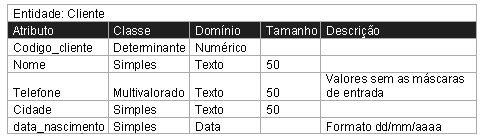
* Um valor de uma chave estrangeira deve corresponder a uma valor existente em uma chave primária associada (ou valor de chave única). Caso contrário, deve ser nulo (NULL).
* Uma chave estrangeira deve referenciar uma chave primária ou uma coluna de chave única.

Conceito de Domínio

Define os tipos de dados e especifica os valores válidos em um campo.



(imagem 15: retirada através da função PrintScreen do Google Imagens dia 06/03/2023)



(imagem 15: retirada através da função PrintScreen do Google Imagens dia 06/03/2023)

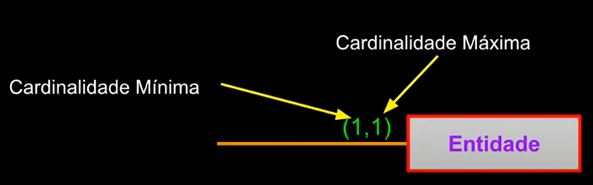
Cardinalidades

Diz respeito ao número de itens que se relacionam nas entidades. Podendo ser:

Cardinalidade Máxima: Trata-se do número máximo de instancias de entidade que podem participar em um relacionamento. Poder ser 1 ou N (muitos).

Cardinalidade Mínima: Número mínimo de instâncias de entidade que devem obrigatoriamente participar em um relacionamento; zero é a participação opcional e um é obrigatória.

Utilizando a notação de Peter Chen:



(imagem 15: retirada através da função PrintScreen do Youtube dia 06/03/2023)

Utilizando os símbolos SQL:

Diagrama, Esquemático

Descrição gerada automaticamente

(imagem 15: retirada através da função PrintScreen do Youtube dia 06/03/2023)

Exemplo:

Diagrama

Descrição gerada automaticamente

(imagem 15: retirada através da função PrintScreen do Youtube dia 06/03/2023)

Obs: Importante realizar a ação lógica de cliente para encomenda e de encomenda para cliente para verificar se a ideia está correta e valida.

Notação de Pé-de-Galinha no Microsoft Visio7

Interface gráfica do usuário

Descrição gerada automaticamente

(imagem 15: retirada através da função PrintScreen do Youtube dia 08/03/2023)

Etapas de desenvolvimento do Banco de Dados

Etapa 1: Regra de Negócio

Analisar a proposta do cliente através da Regra de Negócio que será definida pelo Gerente de Produto. Nessa etapa é necessário especificar e realizar a análise dos requisitos.

Etapa 2: Arquitetura de Três Níveis:

Modelo Conceitual - mundo real por meio de uma visão simplificada dos dados e seus relacionamentos. Assim é possível determinar quais informações necessárias vamos armazenar no Banco de Dados.

1. Descrever tipos de dados requeridos.

2. Seus relacionamentos entre si.

3. Tipos de dados requeridos.

Modelo Lógico - Especificação lógica dos dados em um formato adequado ao SGBD. Os tipos de dados são completamente definidos.

Uma imagem contendo objeto, relógio

Descrição gerada automaticamente

(imagem 16: retirada através da função PrintScreen do Youtube dia 02/03/2023)

Modelo Físico - Detalhamos os componentes de estrutura física do banco de dados, incluindo as tabelas, campos, tipos de valores, restrições etc.

Tabela

Descrição gerada automaticamente

(imagem 17: retirada através da função PrintScreen do Youtube dia 02/03/2023)

Diagrama

Descrição gerada automaticamente

(imagem 18: retirada através da função PrintScreen do Youtube dia 02/03/2023)

Etapa 3: Esquema do Banco de Dados:

Especifica as informações que tratamos durante a Arquitetura do BD. Chamamos de Dicionário de Dados, o documento que armazena todas as informações referentes a cada etapa, nela precisa constar tabelas, visões, funções e demais elementos que venha constar no BD que foi construído.

[Modelo de Dicionário de Dados - SMN](https://github.com/Bakumito/DesafioFinalSQL/blob/main/Documentacao/Dicionario%20de%20Dados%20-%20%20Grupo%2001.pdf)

Integridade de Dados

Manutenção e garantia da consistência e precisão dos dados, sendo um aspecto crítico no design, implementação e uso de sistemas de armazenamento de dados.

A integridade é atingida por meio da aplicação de Restrições de Integridade.

1. Integridade Domínio – Valores inseridos em uma coluna devem sempre obedecer à definição dos valores que são permitidos para essa coluna – os valores do domínio.

São exemplos: Em uma coluna que armazena preços de mercadorias, os valores admitidos são do domínio numérico – ou seja, apenas número.

Integridade de Domínio – Fatores

1. Tipo de Dado no campo
2. Representação interna do tipo de dado
3. Presença ou não do dado
4. Intervalos de valores no domínio
5. Conjuntos de valores discretos

Atributo: *Preço\_Produto*

Domínio: Valor Monetário

* Valor permitido: 25,33
* Valor não permitido: 25 reais e 33 centavos (número e caracteres) -32,33 (valores negativos) vinte e cinco reais e trinta e três centavos (descrição por extenso).

1. Integridade Referencial – Assegura que valores de uma coluna em uma tabela são válidos baseados nos valores em uma outra tabela relacionada.

São exemplos: Se um produto de ID 523 foi cadastrado em uma tabela de Vendas, então um produto com ID 523 deve existir na tabela Produtos relacionada)

Se um registro for excluído em uma tabela, então os registros relacionados em outras tabelas que o referenciam talvez precisem ser excluídos, caso contrário irá ocorrer um erro. O mesmo se dá com atualização de registros.

1. Integridade de Vazio – Informa se a coluna é obrigatória ou opcional – ou seja, se é possível não inserir um valor na coluna.

Uma coluna de chave primária (PK), por exemplo, sempre dever possuir dados inseridos, e nunca pode estar vazia, para nenhum registro.

Valores Nulos (NULL) – Significa que não existem dados. É diferente de zero, espaço, string vazia ou tabulação.

Podem ser problemáticos, pois indicam:

* Valor da coluna não é apropriado;
* Valor não foi inserido;
* Valor é desconhecido;

Interface gráfica do usuário, Aplicativo

Descrição gerada automaticamente

(imagem 18: retirada através da função PrintScreen do Youtube dia 07/03/2023)

1. Integridade de Chave – Os valores inseridos na coluna de chave primária (PK) devem ser sempre únicos, não admitindo-se repetições nesses valores. Os valores de PK também não podem ser nulos. Desta forma, as tuplas (registros) serão sempre distintas.
2. Integridade Definida pelo Usuário – Diz respeito a regras de negócio específicas que são definidas pelo usuário do banco de dados.

São Exemplos: pode-se definir que uma coluna somente aceitará um conjunto restrito de valores.

Dependência Funcional

Uma Entidade com dois atributos que são funcionalmente dependentes entre si, onde um atributo está associado a outro.

Atributo X determina funcionalmente o atributo Y de uma mesma entidade.

São Exemplos:



(imagem 18: retirada através da função PrintScreen do Youtube dia 09/03/2023)

Ex.: O praz o de entrega de um pedido depende do número do pedido considerado.

Logo, o atributo que determina o valor é chamado de Determinante. O outro atributo é chamado de Dependente.

Uma chave primária em uma relação determina funcionalmente todos os outros atributos não-chave na linha.

Dependência Funcional Total

Em uma relação com um PK composta, um atributo não-chave que depende dessa PK como um todo, e não somente de parte dela, é dito como possuindo Dependência Funcional Total.

Interface gráfica do usuário, Diagrama, Aplicativo

Descrição gerada automaticamente

(imagem 18: retirada através da função PrintScreen do Youtube dia 09/03/2023)

No exemplo acima, a entidade Item-Pedido onde constam 3 atributos sendo 2 deles com chave PK onde o 3 atributo possui dependência de 2 atributos PK, logo sendo uma dependência funcional total.

Dependência Funcional Parcial

É quando os atributos não-chave não dependem funcionalmente de toda a PK quando esta for composta. (PK composta diz respeito a duas PK em uma mesma entidade)

Ou seja, existem uma dependência funciona, mas somente de uma parte da chave primária.

Interface gráfica do usuário, Diagrama

Descrição gerada automaticamente

(imagem 18: retirada através da função PrintScreen do Youtube dia 09/03/2023)

No exemplo acima, a entidade de nome Matriculas possui dois atributos PK onde um 3 atributo possui dependência em uma só PK.

Dependência Funcional Transitiva

Ocorre quando um campo não depende diretamente da chave primária da tabela (nem mesmo parcialmente), mas depende de um outro campo não-chave.

Interface gráfica do usuário, Diagrama

Descrição gerada automaticamente

(imagem 18: retirada através da função PrintScreen do Youtube dia 09/03/2023)

No exemplo acima, o atributo Nome\_Vendedor depende funcionalmente do Cód\_Vendedor, que não é chave primária na tabela. Já o campo Prazo\_Entrega depende da PK, Num\_Pedidos.

Dependência Multivalorada

Ocorre quando, para cada valor de um atributo A, existe um conjunto de valores para outros atributos B e C que estão associados a ele, mas são independentes entre si.

Sendo representada da seguinte maneira: A->>B. Onde B é a coluna que depende de A.

Tela de vídeo game

Descrição gerada automaticamente

(imagem 18: retirada através da função PrintScreen do Youtube dia 09/03/2023)

Ano e Cor são independentes entre si e dependem do modelo do carro. Essas duas colunas são dependentes multivalorados do modelo.

Normalização

Consiste em um processo de análise de uma relação para assegurar que seja bem formada.

Decompor relações com anomalias para produzir relações menores e bem estruturados. Ou seja, em uma relação normalizada podemos inserir, excluir ou modificar registros sem criar anomalias.

O processo de normalização aplica a um esquema de relação uma série de testes para certificar que ele satisfaça uma Forma Normal (FN).

O objetivo da normalização é analisar esquemas de relação (tabelas) com base em suas dependências funcionais e chaves primárias para:

1. Minimizar Redundâncias
2. Minimizar anomalias de inserção, exclusão e modificação.

Codd (responsável pelo processo de normalização) propôs originalmente 3 formas normais. Chamamos de 3FN e uma definição mais robusta foi proposta por Boyce e Codd, denominada Forma Normal de Boyce Codd (FNBC).

As relações são decompostas em esquemas de relação menores que atendem aos testes de forma normal (FN).

O ideal é que o projeto do banco de dados relacional alcance a FNBC ou a 3FN para cada tabela.

1. Primeira Forma Normal

Definida historicamente para reprovar atributos multivalorados, compostos e suas combinações.

O domínio de um atributo deve incluir apenas valores atômicos (indivisíveis), e o valor de qualquer atributo em uma tupla deve ser o único valor do domínio desse atributo.