

# **MODELAGEM DE DADOS**

## **FUNDAMENTOS DE BANCOS DE DADOS - UNIDADE 01**

Um banco de dados consiste no conjunto de dados acessíveis aos diversos sistemas de uma determinada empresa. As transações são processos que incluem vários bancos de dados ou somente uma parte deles para realizar consultas em sua base de dados. Um modelo consiste na representação abstrata de um sistema real em um modelo gráfico. Se trata de um meio de comunicação do projeto conceitual de simples compreensão.

## **ELEMENTOS DE UM BANCO DE DADOS**

**HARDWARE (LOCAL DE ARMAZENAMENTO DOS DADOS)**

**SOFTWARE (SISTEMAS DE GERENCIAMENTO DE BANCO DE DADOS)**

**USUÁRIOS (ATORES QUE OPERAM NA BASE DE DADOS)**

## **SISTEMAS GERENCIADORES DE BANCOS DE DADOS (SGBDs)**

Uma das principais razões para usarmos os Sistemas Gerenciadores De Banco De Dados é ter um controle central sobre os dados e sobre os programas que os acessam além de garantir a segurança e a integridade das diversas informações. Esses programas oferecem aos usuários uma representação conceitual conhecida informalmente como abstração de dados que esconde alguns detalhes menos importantes da estrutura do banco de dados.

## **ATOMICIDADE**

Garante que nenhuma ou todas das operações da transação sejam realizadas com sucesso de forma que o sistema mantenha um registro dos antigos valores de quaisquer dados a serem alterados. Caso haja algum problema durante a transação os dados antigos são reestabelecidos como se nunca tivessem sido alterados.

## **CONSISTÊNCIA**

Preserva as regras impostas no banco de dados. Assim que a transação for finalizada todos os dados devem estar íntegros. A consistência é a garantia de manter os dados íntegros durante e depois da transação realizada no banco de dados.

## **ISOLAMENTO**

Garante que uma transação não interfira no trabalho de outra. Novas transações só poderão ser realizadas após o término da transação atual. Alterações feitas por transações simultâneas precisam ser isoladas das alterações feitas por outras simultâneas.

## **DURABILIDADE**

Garante um meio de armazenamento confiável e seguro.

## **SISTEMAS GERENCIADORES DE BANCOS DE DADOS MAIS USADOS**

**ORACLE**

**SQL SERVER**

**MYSQL**

**POSTGREE**

# **BANCO DE DADOS RELACIONAL**

O modelo relacional não é algo estático e se expande assim como a matemática. Sua proposta se baseia na ideia de que as informações e suas relações em uma base de dados podem ser representadas em tabelas. Essas tabelas também podem ser chamadas de entidades que representam objetos com existência física como uma pessoa ou animal e também objetos com existência conceitual como um projeto ou departamento de uma empresa.

## **ADMINISTRADOR DO BANCO DE DADOS (DBA)**

Agente que tem controle total sobre um SGBD.

## **DADOS COMO APOIO A TOMADA DE DECISÃO**

O objetivo de um banco de dados está muito além do simples cadastro de informações que depois serão usadas por outros sistemas. Um banco de dados é considerado por muitas uma mina de ouro e um estudo minucioso do seu conteúdo pode revelar informações preciosas para empresas. Sistemas de apoio à decisão são sistemas que ajudam na análise de informação do negócio visando ajudar os gestores a tomarem decisões estratégicas e a apontar os problemas existentes e futuros que possam ocorrer.

## **DATA WAREHOUSE**

Um Data Warehouse é um arquivo ou repositório de informações obtidas de vários bancos de dados armazenados em um único local. Consiste em um tipo especial de banco de dados orientado por assunto que pode variar com o tempo muito usado na tomada de decisões. Esse modelo surgiu pela necessidade de preservar o banco de dados original de uma empresa.

## **DATA MINING E BUSINESS INTELLIGENCE (BI)**

Data Mining é o processo de analisar grandes bancos de dados de forma semiautomática e de responder a perguntas estratégicas em um período de tempo curto. Esse processo atua em nível mais estratégico para fornecer à empresa o conhecimento acerca do ambiente para decisões em longo prazo. Se refere à mineração ou à descoberta de novas informações em função de regras ou padrões em grandes quantidades de dados que podem ser aplicadas em pesquisas ou empresas visando um aumento significativo dos ganhos. O Business Intelligence consiste no processo de coleta e análise de informações para a gestão de negócios.

## **SEGURANÇA**

Segurança da informação é um tema que ganhou muita notoriedade recentemente. Os ataques internos e externos podem afetar os dados e a própria administração da empresa. Criar regras que garantam segurança aos dados é de fundamental importância. Essas regras são criadas conforme as necessidades de cada companhia e estabelecem critérios de acesso físico e remoto ao banco de dados da empresa. O primeiro ponto da segurança é reforçar a autenticação do usuário e garantir que tenham permissão de acesso somente os trabalhadores previamente autorizados conforme seu nível hierárquico dentro da empresa. Melhorar as diversas senhas que controlam o acesso ao banco de dados também deve ser considerado.

## **MODELOS DE BANCO DE DADOS - UNIDADE 02**

Projetar um banco de dados sem estudo e de forma apressada pode causar vários problemas como um desempenho abaixo do esperado ou resultados equivocados. Um modelo consiste em um detalhamento dos tipos de informações que serão guardadas num determinado banco de dados. Algumas linguagens para modelagem de dados classificadas conforme sua forma de representar modelos podem ser usadas em seu desenvolvimento. Essa forma de representar modelos consiste em um esquema de banco de dados.

# ETAPAS DO DESENVOLVIMENTO DE UM BANCO DE DADOS

## LEVANTAMENTO DE REQUISITOS

O processo de desenvolvimento de um banco de dados sempre começa com o levantamento dos requisitos que consistem nas funcionalidades que o sistema deverá ter. Por meio desse levantamento também é possível ter o domínio do problema que o banco de dados solucionará.

## MODELAGEM CONCEITUAL

Depois do levantamento dos requisitos é gerado um esquema conceitual ou uma visão mais abrangente e geral do banco de dados. O objetivo da modelagem conceitual é o apontamento do problema e não a sua solução. Mostra o que precisará existir no banco de dados de uma forma geral mas sem se preocupar de que forma isso será realizado.

## MODELAGEM LÓGICA

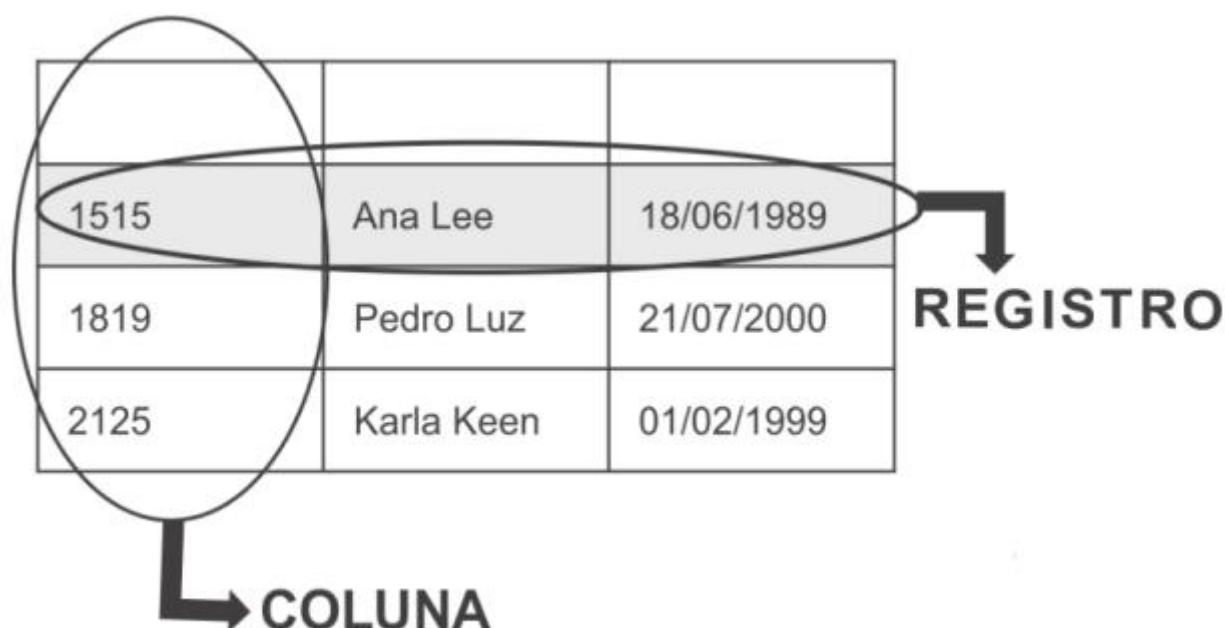
Nesta fase o modelo conceitual se transforma em um modelo lógico com mais detalhes e mais próximo do que será de fato implementado. Os objetos e suas características bem como seus relacionamentos e as chaves entre as tabelas são representados de acordo com as regras e com os limites impostos por um determinado SGBD.

## MODELAGEM FÍSICA

Nesta etapa são determinadas as estruturas de armazenamento interno e os diversos caminhos de acessos à base de dados. Paralelamente são criados os sistemas de aplicação que interagem com a base de dados implementada. As regras de segurança e recuperação também devem ser implementadas bem como as permissões de acesso de cada usuário ao banco de dados.

## MODELO DE ENTIDADE E RELACIONAMENTOS (MER)

O modelo relacional de um banco de dados é criado a partir do levantamento dos requisitos e do modelo conceitual. Uma abordagem relacional parte do princípio de que as informações em uma base de dados devem ser representadas como tabelas. Cada tabela é definida com um conjunto de campos que descrevem suas características. Cada ala representa um conjunto de campos chamados de registros. As conexões entre entidades consistem num relacionamento.



## **TIPOS DE ENTIDADES**

**FORTES (INDEPENDENTE DE OUTRAS ENTIDADES PARA EXISTIR)**

**FRACAS (DEPENDENTE DE ENTIDADES FORTES PARA EXISTIR)**

**AGREGADAS (SE REPETEM EM MAIS DE UMA ENTIDADE)**

**SUBORDINADAS (ATRIBUTOS ESPECÍFICOS DE UMA ENTIDADE)**

**ASSOCIATIVAS (SOMENTE EXISTE DEVIDO AOS RELACIONAMENTOS)**

## **GRAUS DE CARDINALIDADE**

**UM PARA UM (1 - 1)**

- TABELAS SE RELACIONAM COM APENAS UM ELEMENTO DA OUTRA

**UM PARA MUITOS (1 - N)**

- UMA TABELA REFERENCIA DIVERSOS ELEMENTOS DA OUTRA

- ESSA OUTRA PODE REFERENCIAR APENAS UM ELEMENTO DA PRIMEIRA

**MUITOS PARA MUITOS (N - N)**

- AMBAS PODEM REFERENCIAR DIVERSOS ELEMENTOS UMA DA OUTRA

## **DIAGRAMA DE ENTIDADE E RELACIONAMENTO (DER)**

O Diagrama de Entidades e Relacionamentos tem como objetivo central a preparação para o desenvolvimento físico do banco de dados. Essa fase é dependente direta da modelagem lógica e requer a aplicação de comandos para a criação das tabelas e dos campos.

## **TIPOS DE CHAVES**

Em uma determinada base de dados existe uma tabela de clientes com centenas de registros cadastrados. Consequentemente existe também a possibilidade de se procurar um determinado cliente pelo seu nome e aparecer mais de um registro como resultado dessa pesquisa. Para solucionar esse problema se faz necessário estabelecer campos que visam comprovar esses registros armazenados. Alguns valores até poderão ser repetidos mas obrigatoriamente deve haver um campo na tabela que nunca se repete. Esse campo será a chave da tabela.

### **CHAVE PRINCIPAL**

Consiste em algo exclusivo de um registro que nunca se repete na mesma tabela e não pode ter um valor nulo. Podem ser usadas como um índice de referência para os relacionamentos com as demais tabelas do banco de dados.

### **CHAVE COMPOSTA**

Uma chave composta consiste em dois ou mais campos que juntos não se repetem. Diversos campos podem ser usados mas é recomendado que não passem de quatro.

### **CHAVE SURROGADA**

Uma chave surrogada é criada exclusivamente para impedir que os diversos registros de uma determinada tabela venham a se repetir. Nunca pode ser alterada e nem reaproveitada.

### **CHAVE SECUNDÁRIA**

Uma chave secundária é usada para recuperar informações de um cadastro e somente quando não se faz possível usar outras chaves exclusivas de um determinado registro.

## CHAVE ESTRANGEIRA

Uma chave estrangeira é considerada uma chave primária de outra tabela. Por meio dessa chave ocorrem os relacionamentos entre as tabelas de um banco de dados.

## ABORDAGENS (MER) - UNIDADE 03

### ABORDAGENS PARA MODELAGEM DE DIAGRAMAS (MER)

#### TOP-DOWN (BANCOS DE DADOS GRANDES)

- IDENTIFICAR OS CONJUNTOS DE DADOS (PRIMEIRO)
- DEFINIR OS ELEMENTOS DE CADA CONJUNTO (DEPOIS)

#### BOTTOM-UP (BANCOS DE DADOS PEQUENOS)

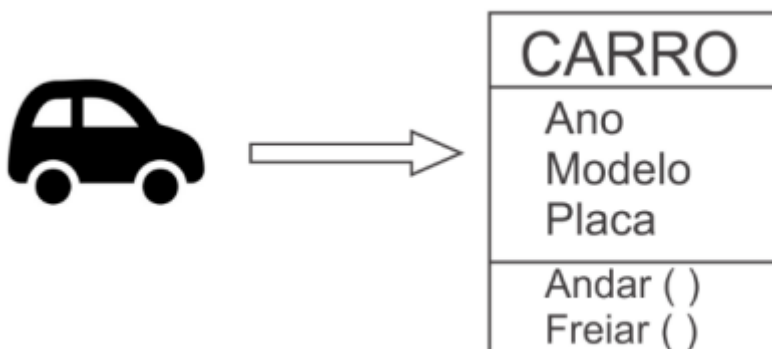
- IDENTIFICAR OS ELEMENTOS (PRIMEIRO)
- AGRUPAR OS ELEMENTOS EM CONJUNTOS (DEPOIS)

## PADRÕES DE DESENVOLVIMENTO

Na maioria dos projetos existe uma grande quantidade de tabelas e campos envolvidos o que torna necessário criar padrões de desenvolvimento. Em uma modelagem onde diversas pessoas trabalham o mesmo campo pode ser referenciado ou criado com nomes diferentes caso não exista um padrão. Criar um dicionário de dados para estabelecer uma padronização e uma documentação sobre cada tabela criada pode ser uma solução para esse problema. Cada empresa tem o seu próprio padrão de desenvolvimento.

## LINGUAGEM DE MODELAGEM UNIFICADA (UML)

Podemos agrupar as informações comuns de um objeto e formar classes que podem ser classificadas como entidades teóricas usadas para modelar objetos e devem ser criadas exclusivamente para guardar seus diversos aspectos. Quando uma classe é criada ela guarda as informações na memória do computador e preenche seus atributos com os dados. Uma classe de persistência tem como objetivo o armazenamento das informações em um meio persistente para que possam ser recuperadas posteriormente.



## PARTES DE UMA CLASSE

**NOME (REFERENTE AO QUE A CLASSE REPRESENTA)**

**ATRIBUTOS (REFERENTE AOS ASPECTOS DO OBJETO)**

**MÉTODOS (REFERENTE AOS COMPORTAMENTOS DO OBJETO)**

# **FUNDAMENTOS DA LINGUAGEM DE MODELAGEM UNIFICADA**

Para ajudar no processo de modelagem de classes existe a UML que tem como seu objetivo padronizar os projetos de sistemas e abranger não somente os aspectos conceituais como as regras do negócio como também as classes e os esquemas do banco de dados além do reuso de componentes que acarreta em grande economia de recursos financeiros.

## **TIPOS DE DIAGRAMAS (UML)**

### **DIAGRAMA DE CLASSES (APRESENTA)**

- **CONJUNTO DE CLASSES E SEUS RELACIONAMENTOS**

### **DIAGRAMA DE OBJETOS (DEMONSTRA)**

- **COMO OS DADOS DO OBJETO FICAM ARMAZENADOS EM CLASSES**

### **DIAGRAMA DE CASO DE USO (APRESENTA)**

- **USUÁRIOS E FUNCIONALIDADE DO SISTEMA**

### **DIAGRAMA DE SEQUÊNCIA (DEMONSTRA)**

- **SINERGIA ENTRE OS OBJETOS PELA ORDEM TEMPORAL**

### **DIAGRAMA DE ATIVIDADES (DETERMINA)**

- **FLUXO DE TAREFAS QUE PODEM SER EXECUTADAS PELO SISTEMA**

### **DIAGRAMA DE ESTADOS (APRESENTA)**

- **CONJUNTO DE ESTADOS DE UM OBJETO**
- **COISAS QUE PODEM MUDAR O ESTADO DO OBJETO**

### **DIAGRAMA DE COMPONENTES (DEMONSTRA)**

- **COMPONENTES DO SISTEMA E SEUS RELACIONAMENTOS**

## **RELACIONAMENTO ENTRE AS CLASSES**

Podemos relacionar as classes através de hierarquias onde elas compartilham suas funções e características comuns. Algumas classes podem receber particularidades exclusivas de outras para permitir um suporte melhor aos relacionamentos. Generalização e especialização são tipos de relacionamentos entre entidades onde determinam que uma entidade superior contenha um ou mais conjuntos de entidades inferiores. Quando agrupamos diversas entidades são em uma entidade genérica temos uma generalização. Quando novas entidades são criadas com atributos que acrescentam detalhes a entidade genérica temos uma especialização.

## **FERRAMENTAS CASE DE MODELAGEM (DER)**

Em grande parte dos projetos de bancos de dados diversos analistas e programadores acabam trabalhando e assim se faz necessário estabelecer padrões de desenvolvimento e criar regras para que todos os envolvidos no projeto modelem da mesma forma. As ferramentas CASE apresentam uma série de serviços que ajudam no desenvolvimento de sistemas e podem ser usadas para automatizar tarefas. Atendem a praticamente todas as fases de um projeto.

### **CÓDIGOS (AUTOMATIZADOS)**

### **DOCUMENTOS (AUTOMATIZADOS)**

### **TESTES (AUTOMATIZADOS)**

### **RELATÓRIOS (AUTOMATIZADOS)**

## **FERRAMENTAS CASE MAIS USADAS**

**RATIONAL ROSE**

**ASTAH**

**GENEXUS**

**MULTICASE**

## **NORMALIZAÇÃO DE DADOS - UNIDADE 04**

A modelagem de dados tem como sua essência o aprimoramento dos processos. Em um projeto de banco de dados o maior problema é a redundância que geralmente só é notada quando o mesmo está em pleno uso pela empresa. Os danos que a redundância pode causar consistem na repetição de informações que ocasiona a duplicidade e possíveis erros em relatórios.

### **REDUNDÂNCIA CONTROLADA**

Por questões de desempenho Do sistema a redundância poderá ocorrer e esse fato é chamado de redundância controlada. Ele é recomendado quando o campo recebe uma grande quantidade de consultas e poucas mudanças. O analista de sistemas deverá decidir se realmente será benéfico deixar a redundância acontecer dependendo do sistema a ser desenvolvido.

### **NORMALIZAÇÃO**

A normalização consiste em uma técnica para avaliar e corrigir estruturas e tabelas ao modo de tornar mínimas as redundâncias de dados e reduzir as chances de problemas. Se trata de um processo rígido e formal que deve ser seguido passo a passo examinando os campos de uma tabela e aplicando determinadas regras com o objetivo de reduzir a redundância e eliminar campos que não dizem respeito a ela.

### **FORMA NORMAL**

Consiste em uma regra que deve ser satisfeita por uma entidade para que ela seja avaliada como uma tabela projetada com exatidão. Existem diversas formas normais com regras rigorosas com o objetivo de averiguar a existência de redundância ou dependências funcionais nas tabelas. Quatro formas normais são consideradas essenciais.

**PRIMEIRA FORMA NORMAL (1FN)**

**SEGUNDA FORMA NORMAL (2FN)**

**TERCEIRA FORMA NORMAL (3FN)**

**QUARTA FORMA NORMAL (4FN)**

### **DEPENDÊNCIA FUNCIONAL**

Podemos descrever a normalização como um processo de organizar os dados em um banco de dados. Para realizar essas tarefas são criadas tabelas e estabelecidos relacionamentos entre elas de acordo com as regras empregadas para proteger os dados e para tornar esse banco de dados mais maleável se adaptando para acabar com redundâncias e dependência inconsistente dos dados que serão armazenados nele. Uma austeridade entre conjuntos de atributos de uma tabela ou relacionamento consiste em uma dependência funcional.

### **TRANSITIVA**

Quando um determinado campo de uma tabela além de depender da chave primária também depende de outro campo ou de outros campos que são integrantes da mesma tabela.

## **TOTAL**

Quando um atributo que não faz parte da chave primária depende diretamente de todos os outros atributos que fazem parte dessa chave primária.

## **PARCIAL**

Quando um campo ou atributo que não faz parte da chave primaria depende somente de alguns dos atributos que fazem parte da chave primária.

## **REQUISITOS (1FN) (2FN) (3FN) (4FN)**

Para adotar as regras de normalização devemos observar os campos de uma tabela.

## **TIPOS DE ATRIBUTOS**

**ATRIBUTO SIMPLES**

**ATRIBUTO COMPOSTO**

**ATRIBUTO MONOVALORADO**

**ATRIBUTO MULTIVALORADO**

**ATRIBUTO DERIVADO**

**ATRIBUTO CHAVE**

## **PRIMEIRA FORMA NORMAL (1FN)**

Para uma tabela atender a 1FN ela deve obedecer a ordem de que seus atributos devem ser simples e não devem ter grupos repetidos ou colunas que possuam mais de um valor. Todos os seus campos ou atributos devem ser monovalorados e atômicos.

## **SEGUNDA FORMA NORMAL (2FN)**

Para uma tabela estar na 2FN ela deve obedecer à regra de que todas as suas colunas que não são chaves dependam exclusivamente de sua chave primária completa. Se faz importante dizer que para uma tabela adotar a 2FN é obrigatório que ela atenda também aos requisitos da 1FN.

## **TERCEIRA FORMA NORMAL (3FN)**

Para uma tabela estar na 3FN todos os seus campos devem ser independentes e não pode haver dependências funcionais além de atender aos requisitos da 2FN. Uma dependência funcional ocorre quando o valor de uma coluna é dependente de outra que não compõe a chave primária de uma tabela. Para atender aos requisitos dessa regra é necessário eliminar todas as dependências transitivas removendo todos os campos dependentes de outras tabelas.

## **QUARTA FORMA NORMAL (3FN)**

Para atender as demandas da 4FN uma tabela precisa que todo campo seja atômico e que não existam dependências multivaloradas além de estar na 3FN. Uma dependência multivalorada ocorre quando as informações inseridas nas tabelas se repetem produzindo redundâncias. Para evitar esse problema o melhor a se fazer é separar essa tabela reduzindo a ocorrência de dependências. Essa não é a última instância da modelagem de um esquema de banco de dados e por isso os refinamentos podem continuar sendo aplicados.