



## Unidade 2

# Bancos de Dados Relacionais e Linguagem SQL

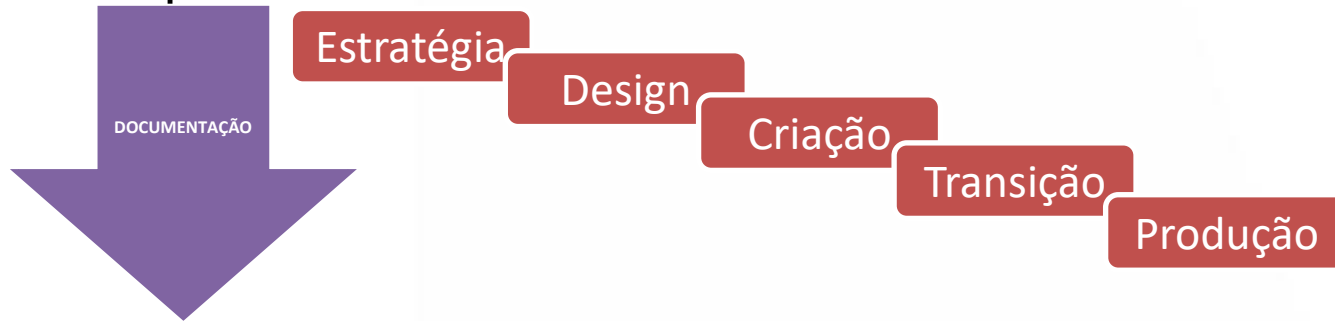
---

## 2.1 Modelos de dados, esquemas e instâncias

---

# Ciclo de vida do desenvolvimento

- Não temos como pensar a construção de um modelo de dados sem ser para atendimento a um sistema, certo?
- Existem passos básicos que podemos seguir para pensar as etapas de desenvolvimento do banco de dados:



# Ciclo de vida do desenvolvimento

- Esta abordagem sistemática *top-down* permite ao desenvolvimento do banco de dados transformar os requisitos de informações de negócio em um banco de dados operacional.
- Conhecer cada estágio deste ciclo, ajuda a planejar melhor um projeto e a ser mais produtivo na construção dos modelos.

# Estratégia e Análise - Atividades

- Estudar e analisar os requisitos de negócios utilizando entrevistas com usuários identificando os requisitos de dados;
- Prever possíveis necessidades futuras do sistema;
- Criar e revisar os modelos conceituais do sistema;
- Criar a representação gráfica do modelo;

# Design - Atividades

- Transformar o modelo desenvolvido na fase de estratégia e análise.
- Mapear entidades para tabelas, atributos para colunas, relacionamentos para chaves estrangeiras e regras de negócios para restrições.

# Criação - Atividades

- Gravar e executar os comandos para criar as tabelas e os objetos de apoio do banco de dados;
- Preencher as tabelas com dados;
- Desenvolver a documentação do usuário, o texto da ajuda e os manuais de operação para auxiliar no uso e na operação do sistema.

# Transição - Atividades

- Conduzir o teste de aceitação do usuário verificando se o que foi desenvolvido atende aos requisitos de negócio;
- Operar de forma paralela e converter dados existentes quando necessário;
- Proceder com as correções apontadas;



# Produção - Atividades

- Implantar o sistema para os usuários;
- Operar o sistema de produção;
- Monitorar o desempenho e refinar a operação do sistema;

# Dados relacionais - Objetivos

- Bancos de dados fazem parte do nosso dia a dia mesmo que nem pensemos nisto rotineiramente.
- **IoT** (*Internet of Things*) são aplicações recentes para armazenamento de dados coletados através de sensores.
- Questões mais tradicionais como compra de passagens aéreas, caixa eletrônico de banco e ligação telefônica estão na lista.

# Então revisemos as regras

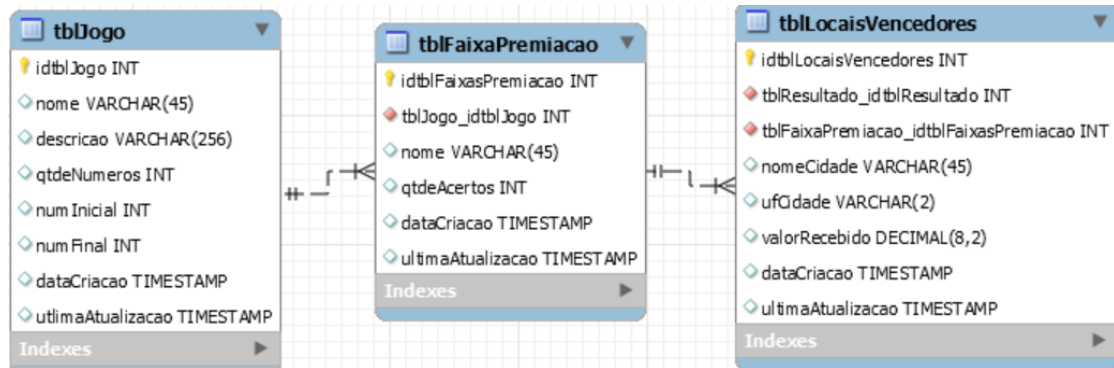
- Após a transformação do modelo conceitual para o modelo físico o banco de dados relacional permite que tabelas sejam relacionadas por meio de um campo comum.
- Se compartilharem um campo comum, apenas duas tabelas já podem ser consideradas um banco de dados relacional.
- Porém os bancos de dados costumam possuir muitas tabelas.

# Então revisemos os conceitos

- **tabela:** estrutura de armazenamento básica
- **coluna:** um tipo de dado em uma tabela
- **linha:** dado para uma instância de tabela
- **campo:** o valor encontrado na intersecção entre uma linha e uma coluna.

# Então revisemos os conceitos

- **chave primária:** o identificador exclusivo de cada linha
- **chave estrangeira:** coluna que se refere a uma coluna de chaves primárias em outra tabela



# Então revisemos as propriedades das tabelas

- **Propriedade 1:** as entradas nas colunas têm um valor único.
- **Propriedade 2:** as entradas nas colunas são do mesmo tipo.
- **Propriedade 3:** cada linha é única (a PK as diferenciara).
- **Propriedade 4:** a sequência das colunas não é significativa.
- **Propriedade 5:** a sequência das linhas não é significativa.
- **Propriedade 6:** cada coluna tem um nome exclusivo.

# Schemas e Instâncias

- Quando nosso diagrama entidade relacionamento é implantado no SGBD (estrutura) ele ainda não possui dados armazenados. Neste momento ele é apenas o projeto e, portanto, é chamado de **esquema** de banco de dados que é **modificado com pouca frequência**.
- Informações começam então a serem inseridas, atualizadas ou apagadas ou seja, gerenciadas pelos mecanismos disponíveis no SGBD e, a partir deste momento, com o banco de dados implantado, todo este conjunto passa a se chamar **instância** de banco de dados.

# Instâncias em nuvem

- No contexto da **computação em nuvem**, quando tivermos que contratar uma **instância** para nossos bancos de dados o conceito envolverá também os **recursos computacionais envolvidos como CPU, memória largura de banda e performance de rede disponíveis**. Observe um exemplo extraído da AWS:

| Modelo       | Número de núcleos | vCPU* | Memória (GiB) | Armazenamento | Largura de banda dedicada do EBS (Mbps) | Performance de rede (Gbps) |
|--------------|-------------------|-------|---------------|---------------|---|----------------------------|
| db.m5.large  | 1                 | 2     | 8             | Somente EBS   | Até 4.750                               | Até 10                     |
| db.m5.xlarge | 2                 | 4     | 16            | Somente EBS   | Até 4.750                               | Até 10                     |



# Finalizando

- Na prática quando formos nos conectar a uma instância de banco de dados temos que fornecer o endereço (IP ou nome DNS) e a porta de conexão.
- O nome do esquema (nome do banco de dados) presente naquela instância deverá ser apontado na conexão e, a partir daí podemos manipular os dados com os comandos SQL.

---

*Now, we cameback to  
toolkit!*

---

# Referências bibliográficas

**MACHADO**, Diego. Normalização em Bancos de Dados, 2015. Disponível em: <<https://medium.com/@diegobmachado/normaliza%C3%A7%C3%A3o-em-banco-de-dados-5647cdf84a12>>. Acesso em: 25 set. 2021.

**MELO**, Izabela Vanessa de Almeida . Armazenamento em nuvem: como preservar documentos digitais?. Disponível em: <<http://www.dsc.ufcg.edu.br/~pet/jornal/maio2011/materias/recapitulando.html> >. Acesso em: 25 set. 2021.

**REIS**, Fabio. Modelagem de Dados - Normalização - Forma Normal de Boyce-Codd, 2018. Disponível em: <<https://www.youtube.com/watch?v=o6mSiTO-vak>>. Acesso em: 25 set. 2021.