

Unidade 4

**Conceitos e técnicas sendo
implementado na prática**



Abertura

Olá, amigo(a) discente! Seja bem-vindo(a)!

A grande demanda por softwares fez com que a forma de se desenvolver sistemas mudasse a necessidade de organizar e armazenar os dados e, assim, disponibilizá-los sempre que necessário em um sistema.

Na aula 01, vamos praticar a modelagem de um banco de dados, passando pelo modelo conceitual até a construção lógica de um banco de dados.

Na aula 02, vamos praticar a construção de alguns comandos em banco de dados que foi construído, atribuindo algumas regras de negócio para com algumas tabelas no banco de dados.

A boa modelagem e construção de um banco de dados é vital para o desenvolvimento de um software com qualidade e segurança, dados são a nova moeda de troca no século XXI, saber cuidar, armazenar e zelar por essa moeda, faz com que os profissionais da tecnologia sejam mais valorizados.

Objetivos

- Aplicar os conceitos de modelagem de banco de dados;
- Aperfeiçoar os conhecimentos para com banco de dados relacionais.

Conteúdo Programático

Aula 01 – Modelagem de banco de dados na prática.

Aula 02 – Aplicando os conceitos de banco de dados relacional.

Modelagem de Banco de Dados na prática

Olá, aluno, tudo bem? Vamos começar agora um estudo muito importante sobre banco de dados avançado. Nesta aula, vamos conhecer e praticar a modelagem, e a construção de um banco de dados relacional.

Com a aplicação correta de um banco de dados, diversas empresas vêm obtendo resultados de sucesso, pois tratar os dados corretamente faz com que um sistema seja mais rígido e preparado para futuras adaptações.

Assim, nesta aula vamos aplicar a modelagem de um banco de dados relacional utilizando o Sistema Gerenciador de Banco de Dados MySQL.

Para descrevermos um banco de dados, é preciso conhecer a realidade do problema que precisa ser solucionado e com isso o tipo de dados que serão armazenados na base com as responsabilidades. Com isso, será utilizado a metodologia de situação problema.

Situação Problema

Figura 1. Situação problema personagem Ana



Fonte: disponível em: br.freepik.com. Acesso em: 28 mar. 2022.

Ana é dona de uma empresa que realiza entregas na cidade de Londrina – PR e precisa de um sistema que organize as entregas de acordo com cada região da cidade, atualmente ele possui 2 entregadores por região, sendo assim, possui 8 colaboradores que realizam entregas dentro da cidade. Normalmente, o pacote é recebido por um entregador e esse mesmo é quem realiza a entrega no destino. Além disso, a proposta é armazenar os dados dos clientes para que no futuro a Ana possua métricas para saber quais são os clientes que mais solicitam o serviço de entrega e até mesmo elaborar campanhas de marketing e fidelização dos clientes.

Para bem modelar um banco de dados, é preciso refletir sobre alguns aspectos e abstrair a necessidade do cliente, se perguntando:

- Quem utilizará o sistema?
- Quais são os dados que precisam ser armazenados?



Videoaula 1

Utilize o QR Code para assistir!

Assista ao vídeo para entender melhor o assunto.



Primeiramente, é preciso listar as entidades, sendo elas:

- Funcionário
- Cliente
- Pacote
- Endereço
- Destino
- Origem

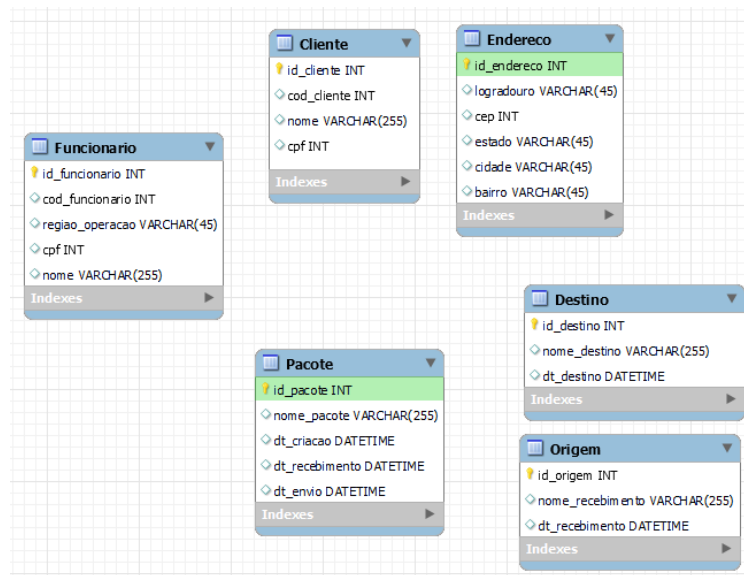
Após realizado esse processo de identificação das entidades é preciso mapear os atributos de cada entidade, para aplicar isso graficamente em um Diagrama de Entidade e Relacionamento. Sendo assim, as entidades são:

Indicação de Vídeo

Como instalar o MySQL + MySQL Workbench no Windows 10.

<https://youtu.be/Pq1i2GX-Hpl>

Figura 2. Entidades com seus atributos



Fonte: O autor, 2022.

Na figura 2, encontra-se o mapeamento gráfico das entidades com seus atributos e tipos, porém durante a modelagem vale aplicar os conceitos iniciais sobre banco de dados relacionais. Dessa forma, pode-se identificar que as entidades **Cliente** e **Funcionário**, tem os mesmos atributos, assim sendo, é plausível aplicar a **Herança** entre as entidades.

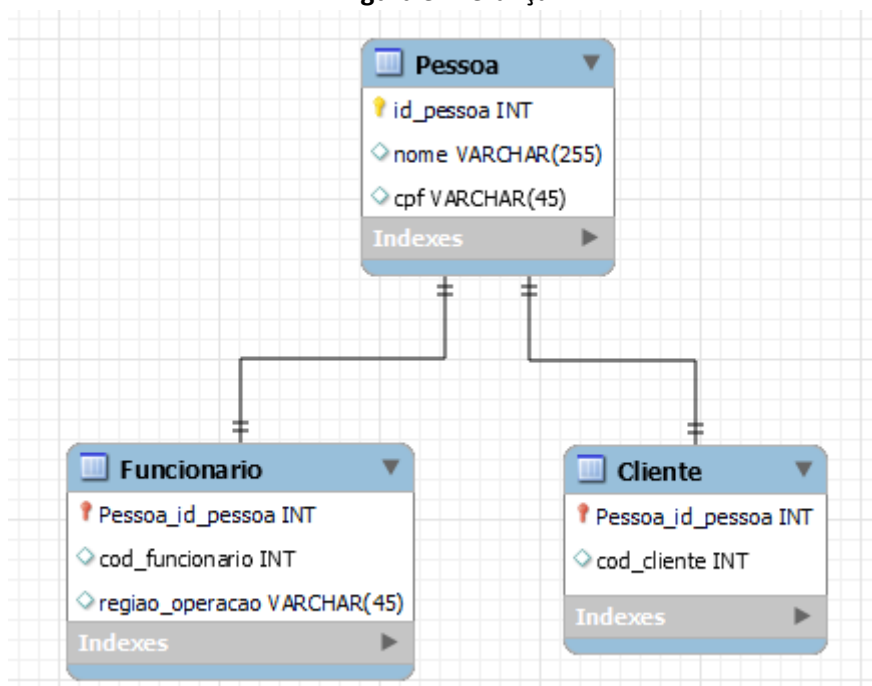
Vale lembrar que podem existir diversas formas e modelos para solucionar a situação problema, o modelo conceitual apresentado é somente um exemplo acadêmico de uma dessas formas e modelos.

Para realizar a modelagem gráfica do BD foi utilizado do *MySQL Workbench* que é uma ferramenta para modelagem e gestão de um banco de dados MySQL.

Indicação de Leitura

MySQL Workbench - <https://www.mysql.com/products/workbench/>

Figura 3. Herança



Fonte: O autor, 2022.

Observando a figura 3, encontra-se a aplicação da herança entre as entidades Cliente e Funcionário que estão herdando os atributos da entidade pessoa, sendo assim, o `id_pessoa` se torna chave primária nas entidades que herdam a entidade pai Pessoa.

Após realizar essa análise, também é necessário aplicar durante a modelagem conceitual qual o relacionamento entre essas entidades, para isso refletir sobre questões como:

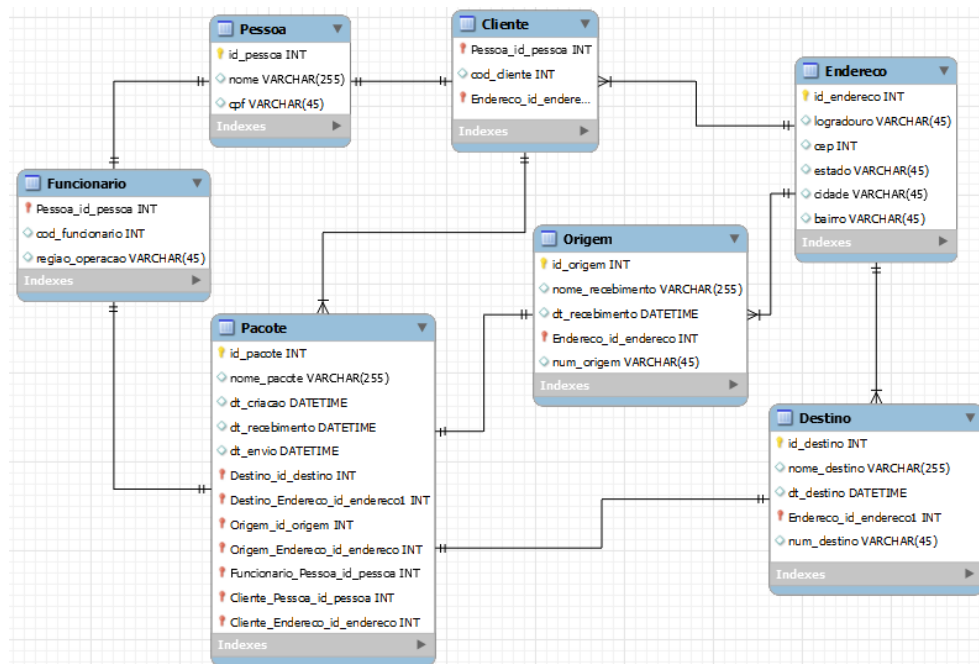
- Como atribuir o relacionamento entre as entidades?
- O que cada entidade pode ou não se relacionar?
- Quais dados essa entidade deve possuir para facilitar o tratamento de dados?
- Como é o comportamento de cada relacionamento?

Indicação de Vídeo

MySQL Workbench e Modelagem de dados.

<https://youtu.be/d07jsZ6aM9Y>

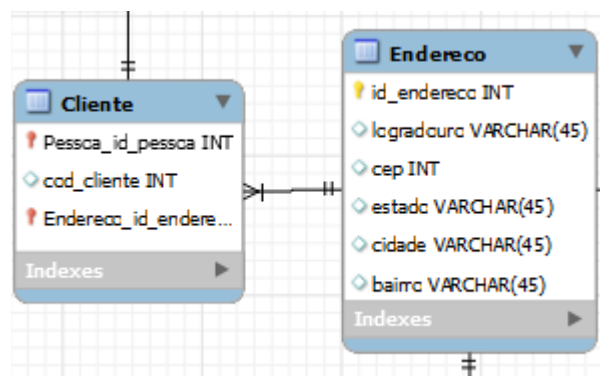
Figura 4. DER situação problema



Fonte: O autor, 2022.

Na figura 4 se observa o Diagrama de Entidade e Relacionamento (DER) da situação problema identificando os relacionamentos entre as entidades presentes. Onde o relacionamento é indicado pela linha que interliga as entidades. Sendo assim estabelecendo as chaves primárias e estrangeiras entre as entidades.

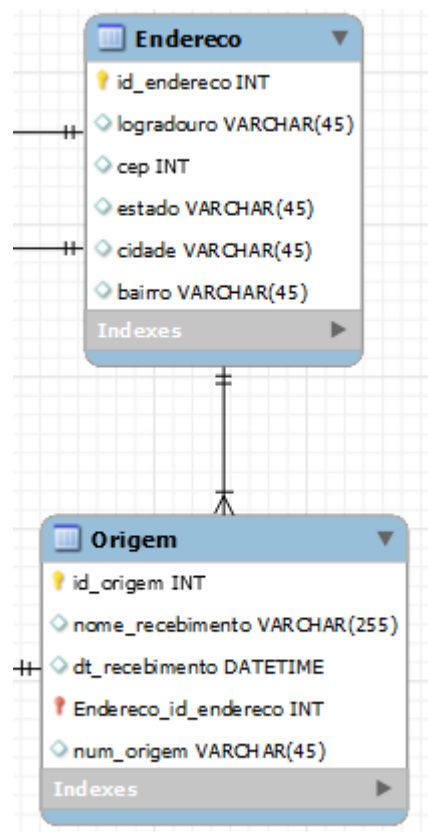
Figura 5. Relacionamento Cliente x Endereço



Fonte: O autor, 2022.

Na figura 5 se encontra a descrição do relacionamento entre as entidades Cliente e Endereço onde o cliente tem um endereço, porém aquele mesmo endereço pode estar presente em vários clientes, resultando em um relacionamento de muitos para um.

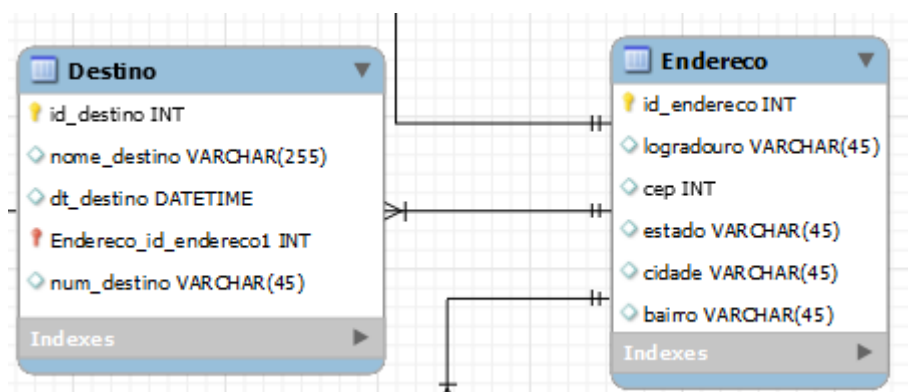
Figura 6. Relacionamento Origem x Endereço



Fonte: O autor, 2022.

Analisando a figura 6 se encontra a entidade Origem com a entidade Endereço, onde a origem pode ter somente um endereço, mas o endereço pode estar em mais de uma origem.

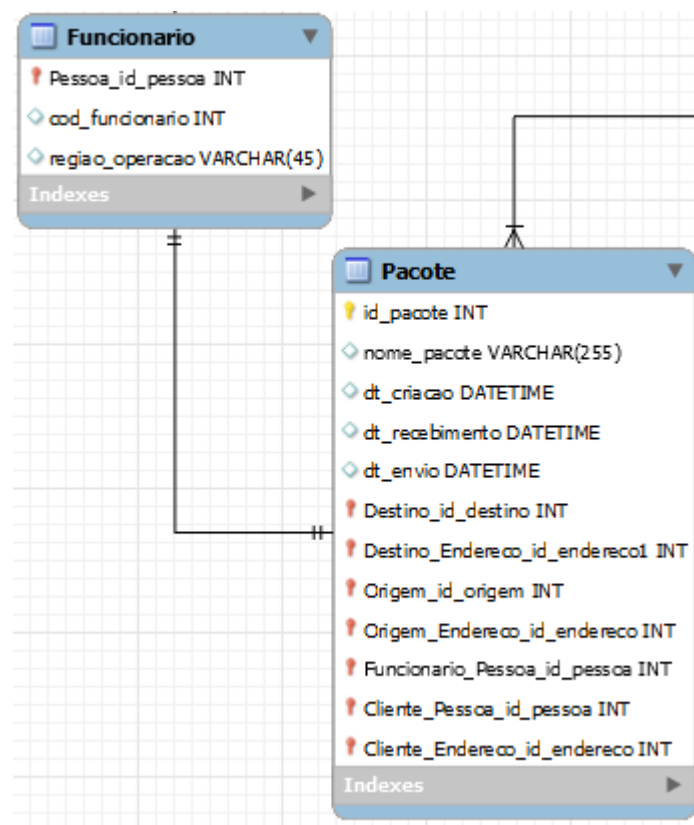
Figura 7. Relacionamento Destino x Endereço



Fonte: O autor, 2022.

Analisando a figura 7 se encontra a entidade Destino com a entidade Endereço, onde a destino pode ter somente um endereço, mas o endereço pode estar em mais de um destino.

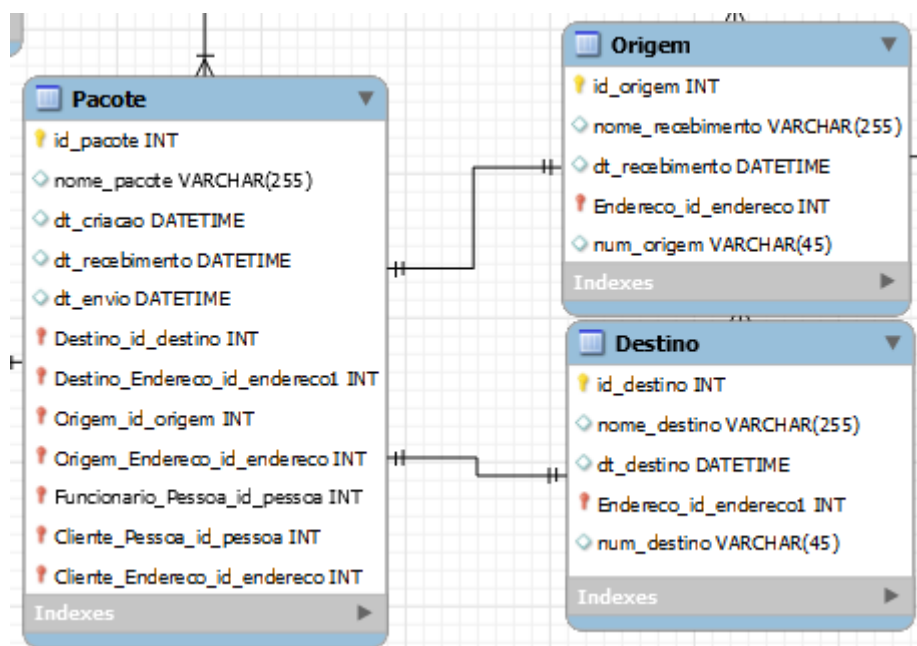
Figura 8. Relacionamento Funcionário x Pacote



Fonte: O autor, 2022.

A figura 8 relata o relacionamento entre as entidades Funcionário e Pacote onde o pacote será recebido e entregue por um funcionário e assim o funcionário vai ser responsável por um pacote até que ele seja entregue.

Figura 9. Relacionamento Destino / Origem x Pacote



Fonte: O autor, 2022.

Analisando a figura 9 se encontra o relacionamento entre as entidades Origem e Destino com a entidade Pacote, onde cada pacote tem somente um destino e uma origem dessa forma o relacionamento é um para um.



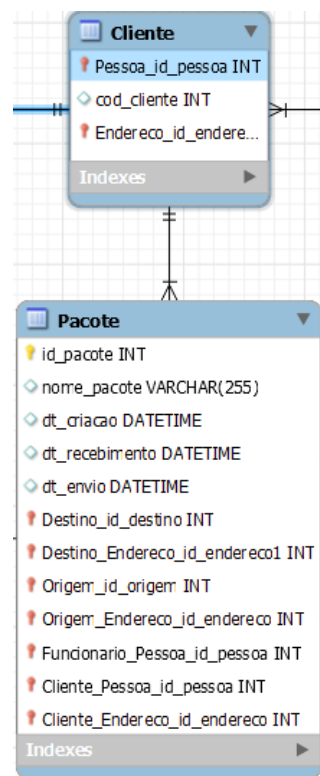
Videoaula 2

Utilize o QR Code para assistir!

Assista ao vídeo para entender melhor o assunto.



Figura 10. Relacionamento Cliente x Pacote



Fonte: O autor, 2022.

Observando a figura 10 se encontra o relacionamento entre a entidade Cliente e Pacote onde um Cliente pode ter vários Pacotes, porém esse Pacote possui somente um cliente, causando um relacionamento um para muitos.

Para facilitar a leitura e construção de um DER é necessário abstrair a necessidade do usuário com uma visão analítica de objetos, visualizando as relações entre os objetos e/ou entidades no banco de dados.

A construção de um modelo conceitual está vinculada a: conhecer, analisar, estruturar, modelar. A solução de um problema real com a necessidade de armazenar dados.



Videoaula 3

Utilize o QR Code para assistir!

Assista ao vídeo para entender melhor o assunto.



Aplicando os conceitos de banco de dados relacional

Olá, estudante, tudo bem? Vamos começar agora um estudo bem diferente sobre banco de dados. Nesta aula, vamos aplicar a modelagem desenvolvida na aula anterior na prática com comandos e aplicando os conceitos na prática. Com isso vamos relembrar alguns conceitos e sua aplicação.

Para aplicar esses conhecimentos é preciso que você assista às videoaulas desta unidade na sequência, pois junto ao docente serão realizadas a construção e alteração dentro do banco de dados.

Modelo Físico

Para a aplicação do modelo físico é preciso voltar a analisar a situação problema para que se possa visualizar algumas regras que podem ser implementadas no momento da modelagem física, para isso é necessário refletir sobre questões como:

- Quais atributos podem ser nulos?
- Quais atributos são obrigatórios?
- Quais atributos devem ser únicos em uma entidade?
- Nesse sistema, quais dados precisam retornar mais rápido?
- Quais *Constraints* serão necessários aplicar?

Em alguns casos no momento de realizar a modelagem lógica se faz útil restabelecer uma conversa com o cliente e até mesmo rever a modelagem conceitual se necessário. A criação do banco de dados é a primeira etapa para assim se discorrer o desenvolvimento do BD, os exemplos que serão apresentados na sequência são aplicados em um banco de dados MySQL.

Figura 11. Create Schema

```
14 • CREATE SCHEMA IF NOT EXISTS `sistema_entregas` DEFAULT CHARACTER SET utf8 ;
```

Fonte: O autor, 2022.

Para realizar a criação do banco de dados é necessário utilizar do comando CREATE SCHEMA indicando a criação de uma nova base de dados, utilizar da clausura IF NOT EXISTS indica que

somente vai criar o banco de dados caso ele ainda não exista. A utilização do DEFAULT CHARACTER SET indica qual tipo de armazenamento será utilizado pelos caracteres.



Videoaula 1

Utilize o QR Code para assistir!

Assista ao vídeo para entender melhor o assunto.



Bom, após aplicar essa prática é o momento de construir o banco de dados, para auxiliar na modelagem lógica de um BD existem recursos na internet como o *SQL Fiddle* - <http://sqlfiddle.com/>.

Utilizando esses recursos se pode testar comandos SQL em diversos modelos de banco de dados. Quando se utilizam recursos como o *SQL Fiddle* não é necessário criar um banco de dados pois a ferramenta já disponibiliza essa função.

Indicação de Vídeo

Testando Comandos SQL

Para seguir a construção lógica do Banco de Dados que chamamos de **sistema_entregas**, vamos precisar analisar os scripts de construção desse banco, sendo eles:

<https://youtu.be/XG7ZTx3YfA>

Tabela Pessoa

Figura 12. Create Table Pessoa

```
19 • CREATE TABLE IF NOT EXISTS `Pessoa` (  
20     `id_pessoa` INT NOT NULL AUTO_INCREMENT,  
21     `nome` VARCHAR(255) NULL,  
22     `cpf` VARCHAR(45) NULL,  
23     PRIMARY KEY (`id_pessoa`),  
24     UNIQUE INDEX `cpf_UNIQUE` (`cpf` ASC));
```

Fonte: O autor, 2022.

A criação da tabela pessoa já possui algumas informações como a identificação de que o **id_pessoa** não pode ser nulo até porque é a chave primária da tabela, além disso ela foi anotada como **AUTO_INCREMENT**, ou seja, automaticamente a base de dados irá realizar o incremento dessa chave.

No caso da tabela pessoa se identifica que o CPF é uma informação única de cada pessoa sendo assim ele utiliza da clausura **UNIQUE** além de ser identificada como index da tabela, o que faz com que uma consulta por cpf na tabela pessoa seja mais otimizada. O **UNIQUE** indica que aquele cpf deve ser único na tabela.

Tabela Funcionário

Figura 13. Create Table Funcionário

```
26 • CREATE TABLE IF NOT EXISTS `Funcionario` (  
27     `Pessoa_id_pessoa` INT NOT NULL,  
28     `cod_funcionario` INT NULL,  
29     `regiao_operacao` VARCHAR(45) NULL,  
30     PRIMARY KEY (`Pessoa_id_pessoa`),  
31     INDEX `fk_Funcionario_Pessoa_idx` (`Pessoa_id_pessoa` ASC),  
32     CONSTRAINT `fk_Funcionario_Pessoa`  
33         FOREIGN KEY (`Pessoa_id_pessoa`)  
34         REFERENCES `Pessoa` (`id_pessoa`)  
35         ON DELETE NO ACTION  
36         ON UPDATE NO ACTION);
```

Fonte: O autor, 2022.

Observando a construção lógica da tabela funcionário é preciso lembrar que esta tabela herda os dados da tabela Pessoa, sendo assim sua chave primária é da tabela Pessoa. Dessa forma, para adicionar um funcionário ele precisa ser uma pessoa obrigatoriamente.

Os índices da tabela nesse caso será a sua chave primária que deve retornar de modo ascendente do maior para o menor.

Tabela Endereço

Figura 14. Create Table Endereço

```
38 • ○ CREATE TABLE IF NOT EXISTS `Endereco` (  
39     `id_endereco` INT NOT NULL AUTO_INCREMENT,  
40     `logradouro` VARCHAR(45) NULL,  
41     `cep` INT NULL,  
42     `estado` VARCHAR(45) NULL,  
43     `cidade` VARCHAR(45) NULL,  
44     `bairro` VARCHAR(45) NULL,  
45     PRIMARY KEY (`id_endereco`));
```

Fonte: O autor, 2022.

Analisando a figura 14 se encontra a criação da tabela Endereço em que armazenará os endereços que podem ser previamente cadastrados.

Dessa forma a chave primária desta tabela é o **id_endereco** que não pode ser nulo e possui o AUTO_INCREMENT. Como não se sabe exatamente como vai ser tratado esse endereço, não foram atribuídas outras regras de validação dentro desta tabela.



Videoaula 2

Utilize o QR Code para assistir!

Assista ao vídeo para entender melhor o assunto.



Tabela Destino

Figura 15. Create Table Destino

```
47 ● ○ CREATE TABLE IF NOT EXISTS `Destino` (  
48     `id_destino` INT NOT NULL AUTO_INCREMENT,  
49     `nome_destino` VARCHAR(255) NULL,  
50     `dt_destino` DATETIME NULL,  
51     `Endereco_id_endereco` INT NOT NULL,  
52     `num_destino` VARCHAR(45) NULL,  
53     PRIMARY KEY (`id_destino`, `Endereco_id_endereco`),  
54     INDEX `fk_Destino_Endereco_idx` (`Endereco_id_endereco` ASC),  
55     CONSTRAINT `fk_Destino_Endereco`  
56         FOREIGN KEY (`Endereco_id_endereco`)  
57         REFERENCES `Endereco` (`id_endereco`)  
58         ON DELETE NO ACTION  
59         ON UPDATE NO ACTION);
```

Fonte: O autor, 2022.

Pode-se observar na figura 15 a criação da tabela Destino, onde essa se relaciona com o endereço onde cada destino tem seu endereço. Dessa forma é gerado o índice para otimização de consultas com o endereço.

Tabela Origem

Figura 16. Create Table Origem

```
65 ● ○ CREATE TABLE IF NOT EXISTS `Origem` (  
66     `id_origem` INT NOT NULL AUTO_INCREMENT,  
67     `nome_recebimento` VARCHAR(255) NULL,  
68     `dt_recebimento` DATETIME NULL,  
69     `Endereco_id_endereco` INT NOT NULL,  
70     `num_origem` VARCHAR(45) NULL,  
71     PRIMARY KEY (`id_origem`, `Endereco_id_endereco`),  
72     INDEX `fk_Origem_Endereco_idx` (`Endereco_id_endereco` ASC),  
73     CONSTRAINT `fk_Origem_Endereco`  
74         FOREIGN KEY (`Endereco_id_endereco`)  
75         REFERENCES `Endereco` (`id_endereco`)  
76         ON DELETE NO ACTION  
77         ON UPDATE NO ACTION);
```

Fonte: O autor, 2022.

A figura 16 demonstra a tabela origem com o seu comando de criação, dessa forma esta possui também um relacionamento com a tabela Endereço onde a origem é um endereço. Sendo assim as tabelas Origem e Destino são relacionadas ao endereço de origem do pacote e o endereço de destino do pacote.

Vale lembrar que as tabelas Origem e Destino possuem um atributo que está relacionado totalmente com a tabela Endereço, que se chama **num_origem** e **num_destino**, que em questão é o número da casa ou apartamento.

Tabela Cliente

Figura 17. Create Table Cliente

```
83 • CREATE TABLE IF NOT EXISTS `Cliente` (  
84     `Pessoa_id_pessoa` INT NOT NULL,  
85     `cod_cliente` INT NULL,  
86     `Endereco_id_endereco` INT NOT NULL,  
87     PRIMARY KEY (`Pessoa_id_pessoa`, `Endereco_id_endereco`),  
88     INDEX `fk_Cliente_Pessoa1_idx` (`Pessoa_id_pessoa` ASC),  
89     UNIQUE INDEX `cod_cliente_UNIQUE` (`cod_cliente` ASC),  
90     INDEX `fk_Cliente_Endereco_idx` (`Endereco_id_endereco` ASC),  
91     CONSTRAINT `fk_Cliente_Pessoa`  
92         FOREIGN KEY (`Pessoa_id_pessoa`)  
93         REFERENCES `Pessoa` (`id_pessoa`)  
94         ON DELETE NO ACTION  
95         ON UPDATE NO ACTION,  
96     CONSTRAINT `fk_Cliente_Endereco`  
97         FOREIGN KEY (`Endereco_id_endereco`)  
98         REFERENCES `Endereco` (`id_endereco`)  
99         ON DELETE NO ACTION  
100        ON UPDATE NO ACTION);
```

Fonte: O autor, 2022.

Ao analisar a figura 17 se encontra a tabela Cliente com seu script de criação, onde essa (assim como a tabela Funcionário), possui um relacionamento com a tabela Pessoa, onde se encontram seus atributos genéricos.

Tabela Pacote

Figura 18. Create Table Pacote

```
1 CREATE TABLE IF NOT EXISTS `Pacote` (  
2   `id_pacote` INT NOT NULL AUTO_INCREMENT,  
3   `nome_pacote` VARCHAR(255) NULL,  
4   `dt_criacao` DATETIME NULL,  
5   `dt_recebimento` DATETIME NULL,  
6   `dt_envio` DATETIME NULL,  
7   `Destino_id_destino` INT NOT NULL,  
8   `Destino_Endereco_id_endereco` INT NOT NULL,  
9   `Origem_id_origem` INT NOT NULL,  
10  `Origem_Endereco_id_endereco` INT NOT NULL,  
11  `Funcionario_Pessoa_id_pessoa` INT NOT NULL,  
12  `Cliente_Pessoa_id_pessoa` INT NOT NULL,  
13  `Cliente_Endereco_id_endereco` INT NOT NULL,  
14  PRIMARY KEY (`id_pacote`),  
15  `Destino_id_destino`,  
16  `Destino_Endereco_id_endereco`,  
17  `Origem_id_origem`,  
18  `Origem_Endereco_id_endereco`,  
19  `Funcionario_Pessoa_id_pessoa`,  
20  `Cliente_Pessoa_id_pessoa`,  
21  `Cliente_Endereco_id_endereco`),  
22  INDEX `fk_Pacote_Destino1_idx` (`Destino_id_destino` ASC, `Destino_Endereco_id_endereco` ASC),  
23  INDEX `fk_Pacote_Origem1_idx` (`Origem_id_origem` ASC, `Origem_Endereco_id_endereco` ASC),  
24  INDEX `fk_Pacote_Funcionario1_idx` (`Funcionario_Pessoa_id_pessoa` ASC),  
25  INDEX `fk_Pacote_Cliente1_idx` (`Cliente_Pessoa_id_pessoa` ASC, `Cliente_Endereco_id_endereco` ASC),  
26  CONSTRAINT `fk_Pacote_Destino`  
27  FOREIGN KEY (`Destino_id_destino`, `Destino_Endereco_id_endereco`)  
28  REFERENCES `Destino` (`id_destino`, `Endereco_id_endereco`)  
29  ON DELETE NO ACTION  
30  ON UPDATE NO ACTION,  
31  CONSTRAINT `fk_Pacote_Origem`  
32  FOREIGN KEY (`Origem_id_origem`, `Origem_Endereco_id_endereco`)  
33  REFERENCES `Origem` (`id_origem`, `Endereco_id_endereco`)  
34  ON DELETE NO ACTION  
35  ON UPDATE NO ACTION,  
36  CONSTRAINT `fk_Pacote_Funcionario`  
37  FOREIGN KEY (`Funcionario_Pessoa_id_pessoa`)  
38  REFERENCES `Funcionario` (`Pessoa_id_pessoa`)  
39  ON DELETE NO ACTION  
40  ON UPDATE NO ACTION,  
41  CONSTRAINT `fk_Pacote_Cliente`  
42  FOREIGN KEY (`Cliente_Pessoa_id_pessoa`, `Cliente_Endereco_id_endereco`)  
43  REFERENCES `Cliente` (`Pessoa_id_pessoa`, `Endereco_id_endereco`)  
44  ON DELETE NO ACTION  
45  ON UPDATE NO ACTION);
```

Fonte: O autor, 2022.

A figura 18 demonstra o script de criação da tabela Pacote, esta tabela possui relacionamento com quase todas as outras tabelas do Banco de Dados, onde o pacote se relaciona com as tabelas Origem, Destino, Funcionário e Cliente. Com isso a tabela Pacote completa a base de dados.



Videoaula 3

Utilize o QR Code para assistir!

Assista ao vídeo para entender melhor o assunto.



A boa aplicação da construção de um banco de dados, auxilia totalmente a implementação de um sistema, a maneira que os dados estão armazenados e a forma que eles podem ser acessados afetam diretamente o comportamento de um sistema.

Em um banco de dados podem existir diversas soluções e modelos diferentes que solucionam o armazenamento e tratamento dos dados, porém cabe ao analista avaliar e encontrar qual a melhor forma de implementar um banco de dados.

Implementar corretamente os conceitos avançados de banco de dados irá facilitar o desenvolvimento e a organização das informações dentro de um sistema. Sendo assim, para construir um banco de dados é necessário conhecer o mercado ao qual o cliente/solicitante trabalha, quais são os aspectos únicos e os dados mais importantes.

Dessa forma a aplicação da modelagem de banco de dados juntamente com as boas práticas de desenvolvimento de software podem ocasionar em um sistema sólido e eficaz que pode durar vários anos.

Atividades Avaliativas e de Fixação

Prezado(a) aluno(a)!

Chegou a hora de colocarmos em prática os conhecimentos adquiridos até o momento!

As avaliações nos possibilitam perceber se estamos obtendo bons resultados e quais são os conteúdos que requerem nossa atenção.

Clique em “**Módulos**” no “**Menu Lateral**” e acesse a **Atividade Fixação e Atividade Avaliativa**.

Atente-se aos prazos! Após a data limite, a avaliação será encerrada. Bons estudos!



Videoaula Encerramento

Utilize o QR Code para assistir!

Assista agora ao vídeo de encerramento de nossa disciplina.



Encerramento

Chegamos ao final da nossa quarta unidade, onde você teve contato com os conceitos práticos de banco de dados. Com certeza, muito do que foi mostrado aqui você já tinha um pequeno conhecimento ou até mesmo já havia utilizado em seu cotidiano. Espero que você tenha compreendido um pouco mais sobre banco de dados avançados.

No mundo dos softwares há muitos dados e formas de armazenar esses dados, porém quando realizado uma boa análise a possibilidade de sucesso aumenta. Na aula 1 você conheceu e foi instigado a praticar a modelagem conceitual de um banco de dados, com uma situação problema.

Na aula 2 foi discutido e implementado o banco de dados que foi conceituado na aula 1, além de testar e construir isso em uma ferramenta web, implementando alterações e até boas

práticas. Juntando as duas aulas teremos conhecimento prático de como alguns conceitos de banco de dados podem ser implementados. Caro aluno, pense um pouco como os conhecimentos desta unidade podem te ajudar nos desafios do seu cotidiano. Até a próxima.

Encerramento da Unidade

Olá aluno, tudo bem? Durante o decorrer da disciplina de banco de dados avançados conseguimos discutir sobre como os conceitos e suas práticas, auxiliam na elaboração de um banco de dados e a manipulação desses dados.

A boa manipulação dos dados interfere diretamente no funcionamento de um sistema, um banco de dados bem elaborado e com a organização correta dos dados, aumentam a segurança e a privacidade de um sistema. Sendo assim agregando maior valor para o cliente.

Claro que a aplicação dos conhecimentos apresentados não serão a solução de todas as adversidades que permeiam o mundo de banco de dados, porém, estar preparados para mudanças é necessário.

Com os conhecimentos obtidos nesta disciplina espera-se que você seja capaz de analisar, identificar, estruturar e aplicar a manutenção dos dados da melhor forma possível, de acordo com a necessidade de cada cliente.

Referências

PRESSMAN, Roger S. **Engenharia de Software**. 1. ed. São Paulo: Person, 2011.

ORACLE, Corporation and/or its affiliates. Documentação MySQL. Disponível em: <https://dev.mysql.com/doc/>. Acesso em: 11fev. 2022.



UNIFIL.BR