

# WebDev - Módulo 4

## Roteiro de aula

### Aula 4 - Hard: Ó o pesado!



#### Tópicos da aula:

- CREATE e DROP TABLE
- INSERT e SELECT



#### Objetivos da aula:

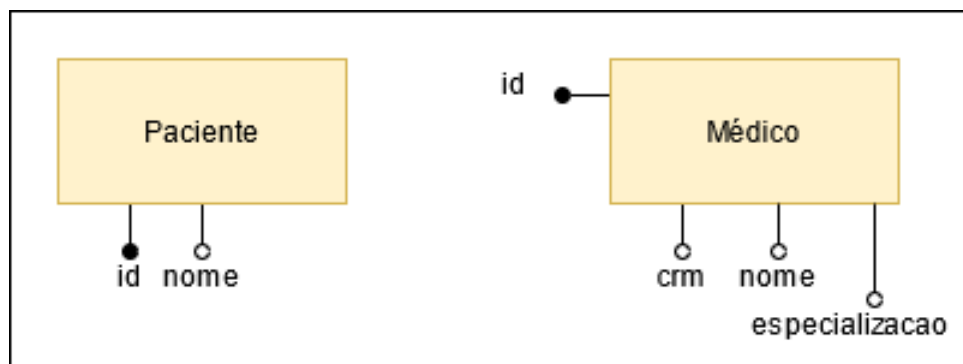
1. Compreender o conceito de commits e rollbacks;
2. Visualizar a relação do conceito de CRUD com banco de dados e a operação CREATE;
3. Realizar a importação de arquivos CSV em um banco de dados via Workbench;



#### Atividades da Aula

##### → Atividade: De volta ao consultório

- ◆ Vamos voltar ao consultório do Dr. Pentescopu e criaremos um banco chamado **consultorio**. Vocês devem criar as tabelas do diagrama que está abaixo! Mais uma vez, não vamos nos preocupar com os relacionamentos.



**Dica:** usem o dbDiagram!



#### Momento Aprendizagem em Pares

- Esse momento é dedicado para vocês desenvolverem suas demandas e entregas para o curso.
- Utilize esse tempo da maneira que preferir, mas atente-se às aulas que você deve realizar as entregas.
  - ◆ **Dica:** Nas Propostas dos projetos, vocês encontram uma sugestão de organização para a realização das atividades.
  - ◆ **Lembre-se:** O momento de Aprendizagem em Pares é justamente para fazer trocas e aprender em comunidade, aproveite seus colegas e se desenvolvam juntos!
- **Entregas:**
  - ◆ Projeto individual: aula 5 - HARD
  - ◆ Projeto em grupo: aula 10 - HARD
  - ◆ Apresentação do projeto: aula 10 - HARD



## Revisão da aula

- O aspecto mais importante de um banco de dados é a capacidade de armazenar dados e a capacidade de manipular dados. **COMMIT** e **ROLLBACK** são duas dessas palavras-chave usadas para armazenar pedidos e reverter o processo de armazenamento de dados. Essas palavras-chave geralmente são usadas no contexto de uma transação.
- O **COMMIT** transaction indica o fim normal da transação. O que tiver de comando depois, já não fará parte desta transação. Neste momento tudo o que foi manipulado passa a fazer parte do banco de dados normalmente e operações diversas passam a enxergar o que foi feito.
- O **ROLLBACK** transaction também fecha o bloco da transação e é a indicação que a transação deve ser terminada, mas tudo que tentou ser feito deve ser descartado porque alguma coisa errada aconteceu e ela não pode terminar normalmente. Nada realizado dentro dela será mantido no banco de dados.
- Essas são as **operações de CRUD**
  1. **Create** operations, que envolvem a gravação de dados no banco de dados
  2. **Read** operations, que consultam um banco de dados para recuperar dados dele
  3. **Update** operations, que alteram dados que já existem em um banco de dados

4. **Delete** operations, que removem dados permanentemente de um banco de dados

→ A **API** contém o código e os métodos. O banco de dados armazena e ajuda o usuário a recuperar as informações, enquanto a interface do usuário ajuda os usuários a interagir com o aplicativo. Você pode criar um aplicativo CRUD com qualquer uma das linguagens de programação existentes. E o aplicativo não precisa ser full stack – você pode criar um aplicativo CRUD com JavaScript do lado do cliente.

→ O **MySQL Workbench** é uma ferramenta unificada de design de banco de dados relacional de plataforma cruzada e código aberto que adiciona funcionalidade e facilidade ao trabalho de desenvolvimento MySQL e SQL

→ Como **importar arquivos CSV** em um banco de dados. Abra o MySQL Workbench e veja as instruções: crie um esquema e depois selecione “Create Table” ou “Table Data Import Wizard” para criar uma tabela dentro do esquema. Para importar um arquivo CSV encontre o caminho do arquivo CSV que você deseja importar. Encontre o caminho para os dados. Agora você pode criar uma nova tabela ou adicionar a uma tabela existente.



## Para ajudar



### Links interessantes:

- Como conectar no [MySQL via Workbench](#)