



**ESCOLA ESTADUAL GOVERNADOR MILTON  
CAMPOS**

Criado pelo Regulamento n° 27, de 05/02/1854 - Portaria n° 477/2001,  
de 13/06/2001 Rua Fernandes Tourinho, 1020 –

Lourdes – Belo Horizonte - MG

CEP 30112-000 – Tel./fax (31) 2510-9779 – e-mail:  
[escola.2135@educacao.mg.gov.br](mailto:escola.2135@educacao.mg.gov.br)

I

-

## Trabalho de Banco de Dados

TURMA :204

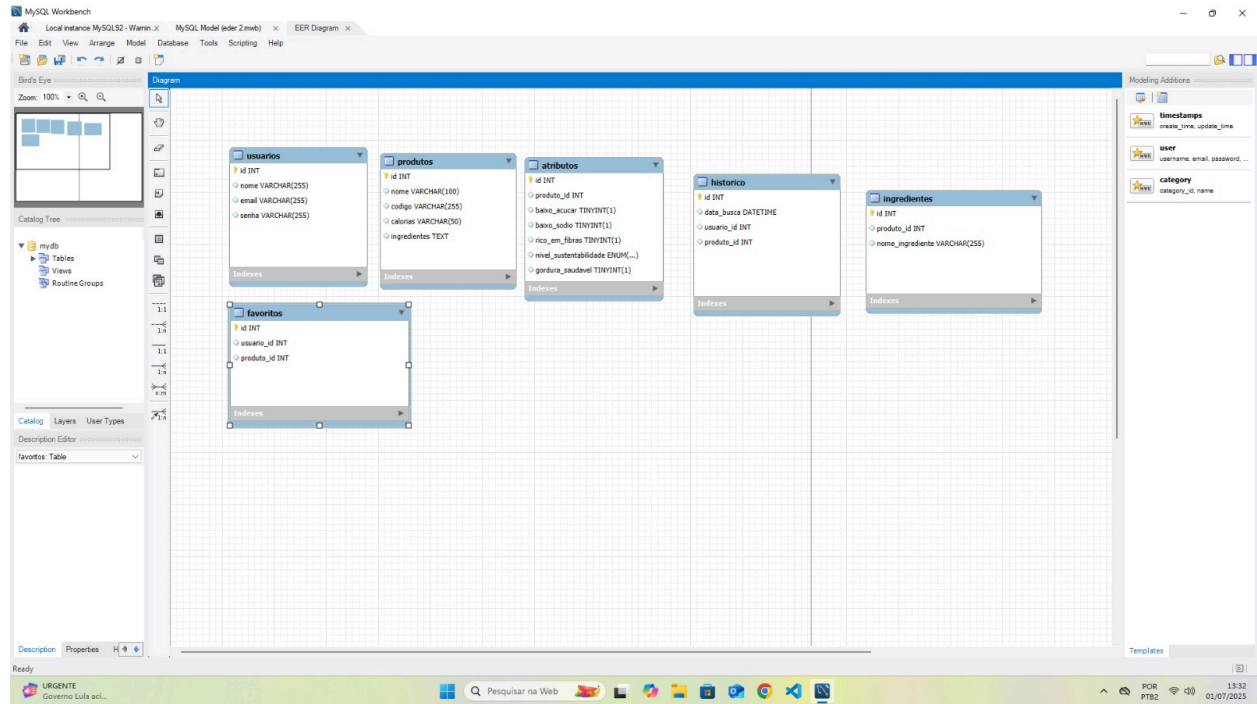
Nome dos integrantes: **Isaac Matos de Souza**

Rebeca Duarte Arcanjo

José Tadeu Silva Viana da Conceição

Eder Henrique J da silva

Miguel Arcanjo Delogo



```
Limit to 1000 rows
1 • CREATE TABLE produtos (
2     id INT AUTO_INCREMENT PRIMARY KEY,
3     codigo VARCHAR(100),
4     nome VARCHAR(255),
5     calorias VARCHAR(50),
6     fabricante VARCHAR(100),
7     codigo_barras VARCHAR(100) UNIQUE,
8     alto_teor_acucar BOOLEAN DEFAULT FALSE,
9     observacoes TEXT,
0     data_cadastro DATETIME DEFAULT CURRENT_TIMESTAMP
1 );
2
3
4
5 • CREATE TABLE ingredientes (
6     id INT AUTO_INCREMENT PRIMARY KEY,
7     produto_id INT NOT NULL,
8     nome_ingredientes VARCHAR(255) NOT NULL,
9     FOREIGN KEY (produto_id) REFERENCES produtos(id)
0     ON DELETE CASCADE
1 );
2
3
4
5 • INSERT INTO produtos (nome, calorias, fabricante, ingredientes)
6     VALUES ('Granola Natural', 120, 'Granola Vida', 'Aveia, mel, castanhas');
7
8
```

De início do trabalho a ideia é bem simples: montar um banco de dados que ajuda a organizar informações de alimentos, ingredientes, calorias e umas paradas que têm a ver com saúde e bem-estar tipo saber o que você tá comendo

- ideia principal:

O foco é montar uma base de dados onde eu consiga guardar informacoes de produtos alimentares. Tipo nome, calorias, código de barras, se tem açúcar demais, e por aí vai. Serve pra ajudar quem quer se alimentar melhor ou entender melhor o que tá consumindo.

- MYSQL para criar o banco precisamos utilizar a primeira linha de código q é:

CREATE DATABASE nome do banco q é (nutriscan); e para usarmos a database que criamos ou ja esta criada, utilizamos o USE e o nome do banco q é (nutriscan);  
Aí com isso eu já estava dentro do banco e comecei a montar as tabelas.

- Tabela dos produtos

Criei a tabela produtos, onde eu guardo:

nome do produto,

código,

calorias,

fabricante,

código de barras,

se tem alto teor de açúcar ou não (trabalhando com as respostas (sim ou não)),

e a data que foi cadastrado.

- O código que usamos foi esse:

```
CREATE TABLE produtos (  
  id INT AUTO_INCREMENT PRIMARY KEY,  
  codigo VARCHAR(100),  
  nome VARCHAR(255),  
  calorias VARCHAR(50),  
  fabricante VARCHAR(100),  
  codigo_barras VARCHAR(100) UNIQUE,  
  alto_teor_acucar BOOLEAN DEFAULT FALSE,  
  observacoes TEXT,  
  data_cadastro DATETIME DEFAULT CURRENT_TIMESTAMP  
);
```

oque rola aqui:

Esse código cria a tabela produtos, onde ficam as infos principais dos alimentos. O id identifica cada produto e vai aumentando sozinho. O código e o nome são pra identificar o produto. calorias e fabricante falam quantas calorias tem e quem fez. código\_barras não pode repetir, porque é único pra cada produto. alto\_teor\_acucar marca se o produto tem muito ou pouco açúcar (e ele vem falso por padrão). o observacoes serve pra anotar qualquer coisa extra. E o data\_cadastro já pega a data e hora que foi cadastrado, sozinho. Tudo organizado pra depois ligar com os ingredientes.

- Tabela dos ingredientes:

Pra não ficar aquela bagunça de colocar os ingredientes tudo junto na mesma tabela, eu criei uma tabela separada chamada ingredientes.

Ela é ligada com a de produtos pelo produto\_id, e é assim:

```
CREATE TABLE ingredientes (  
  id INT AUTO_INCREMENT PRIMARY KEY,  
  produto_id INT NOT NULL,  
  nome_ingrediente VARCHAR(255) NOT NULL,  
  FOREIGN KEY (produto_id) REFERENCES produtos(id)  
ON DELETE CASCADE  
);
```

o que rola aqui:

Essa tabela é pros ingredientes dos produtos. O id identifica cada ingrediente. O produto\_id liga esse ingrediente a um produto da outra tabela. O nome\_ingrediente é só o nome mesmo, tipo "Açúcar".

O FOREIGN KEY faz a conexão com a tabela produtos, E o ON DELETE CASCADE é tipo: se eu apagar um produto, os ingredientes ligados a ele já somem junto, tipo um efeito dominó.

Coca-Cola:

Pra testar isso, eu cadastrei a Coca-Cola como exemplo assim:

```
INSERT INTO produtos (codigo, nome, calorias)  
VALUES ('C001', 'Coca-Cola', '140');
```

mandei os ingredientes dela separados:

```
INSERT INTO ingredientes (produto_id, nome_ingrediente)  
VALUES  
(1, 'Água gaseificada'),
```

```
(1, 'Açúcar'),  
(1, 'Corante caramelo IV'),  
(1, 'Ácido fosfórico'),  
(1, 'Aromatizantes naturais'),  
(1, 'Cafeína');
```

E pronto a Coca ficou com os ingredientes organizados

Se for olhar no diagrama MySQL Workbench, tu vai ver a tabela produtos ligada com a ingredientes por uma setinha, bem simples

E para depois a gente esta pensando em fazer umas consultas brabas, tipo:

“me mostra os produtos com mais calorias”, ou “me mostra os que têm açúcar” e etc.