

2025

# Sistem de gestionare a restaurantelor

BAZĂ DE DATE

AMARIEI ALEXANDRA  
CRIVEANU ANDRA-MARIA  
IONIȚĂ-MITRAN IOANA  
LĂMBUȚĂ GEORGETA RALUCA  
REUȚ DENISA-GEORGETA  
ECHIPA 12 | 423C



## Cuprins

Scopul bazei de date .....	3
Descrierea entităților și asocierilor .....	3
Tabele Principale .....	3
Restaurante .....	3
Produse .....	3
Clienți .....	4
Rezervări .....	4
Comenzi .....	4
Tabele suplimentare rezultate din asocierile N:M .....	4
ProdusComezi .....	4
ProdusRestaurant .....	4
Asocieri .....	5
1. Asociere 1:N .....	5
2. Asocieri N:M .....	5
Schemă relațională .....	6
Model Entitate → Asociere .....	6





## Scopul bazei de date

Baza de date restauranteDB a fost concepută pentru a gestiona informații despre restaurante, produse clienți, comenzi, rezervări. Aceasta permite administrarea eficientă a datelor legate de activitatea unui lanț de restaurante sau a unei platforme de rezervări și livrări.

## Descrierea entităților și asocierilor

- **Restaurante** (tblRestaurante): conține informații despre restaurante (nume, oraș, tip bucătărie, disponibilitate livrare).
- **Produs** (tblProdus): articolele disponibile în Produsurile restaurantelor, legate de restaurant prin cheie externă.
- **Clienți** (tblCienti): informații despre clienți (nume, email, data înregistrării).
- **Rezervări** (tblRezervari): legătura dintre clienți și restaurante pentru o anumită dată/oră și număr de persoane.
- **Comenzi** (tblComenzi): comenzile plasate de clienți, conțin tipuri de produse din tabela Produs.
- **ProdusComenzi** (tblProdusComenzi): numărul de produse dintr-o comandă.
- **ProdusRestaurant** (tblProdusRestaurant): numărul de feluri dintr-un restaurant.

## Tabele Principale

### Restaurante

- `restaurant_id` – ID unic pentru fiecare restaurant (cheie primară).
- `nume_restaurant` – Numele restaurantului.
- `oras` – Orașul unde se află restaurantul.
- `tip_bucatarie` – Specificul culinar (ex: italiană, românească etc.).
- `livrare_disponibila` – Valoare booleană: TRUE/FALSE pentru livrare.
- **Constrângere:** `oras` nu poate fi NULL (`CHECK (oras IS NOT NULL)`).

### Produse

- `produs_id` – ID unic pentru fiecare produs (cheie primară).
- `nume_produs` – Numele produsului.
- `Pret_produs` – Prețul (tip DECIMAL(6,2) – max 9999.99).
- `categorie` – Categorie produsului (ex: fel principal, desert).





## Clienți

- `client_id` – ID unic pentru fiecare client.
- `nume_client` – Numele clientului.
- `email` – Adresa de email (trebuie să fie unică).
- `data_inregistrare` – Data înregistrării în sistem.

## Rezervări

- `rezervare_id` – ID unic pentru fiecare rezervare.
- `client_id` – Clientul care a făcut rezervarea.
- `restaurant_id` – Restaurantul la care s-a făcut rezervarea.
- `data_ora` – Data și ora rezervării.
- `numar_persoane` – Numărul de persoane pentru rezervare.
- `status_rezervare` – Starea rezervării: confirmata, anulata, in\_asteptare (tip ENUM).



## Comenzi

- `comanda_id` – ID unic pentru comandă.
- `client_id` – Clientul care a comandat.
- `restaurant_id` – restaurantul de unde se comandă
- `data_comanda` – Data comenzii.
- `tip_plata` – Modalitatea de plată: cash, card (tip ENUM).
- `Status_comanda` – Starea comenzii: finalizata, in\_pregatire (tip ENUM).

## Tabele suplimentare rezultate din asocierile N:M

### ProdusComezi

- `comanda_id` – Leagă produsul de o anumită comandă.
- `produs_id` – Produsul inclus în comandă.
- `cantitate` – Câte bucăți din acel produs s-au comandat.
- O comandă poate conține mai multe produse.
- Un produs poate apărea în mai multe comenzi.
- Dacă se șterge o comandă sau un produs, legăturile se șterg și ele automat.

### ProdusRestaurant

- `restaurant_id` – Leagă produsul de un restaurant.
- `produs_id` – Identifică produsul asociat.
- `nr_feluri` – Numărul de variante sau feluri ale acelui produs în acel restaurant
- Fiecare produs poate exista în mai multe restaurante.
- Fiecare restaurant poate avea mai multe produse.
- Când un restaurant sau produs se șterge, rândurile din acest tabel se șterg automat (ON DELETE CASCADE).





## Asocieri:

### 1. Asociere 1:N

→ O înregistrare dintr-un tabel A poate fi asociată cu mai multe înregistrări dintr-un tabel B, dar fiecare înregistrare din tabelul B este asociată cu o singură înregistrare din tabelul A.

#### a. **tblClienti -> tblRezervari**

- Un client poate face mai multe rezervări
- Legătură: `tblRezervari.client_id → tblClienti.client_id`

#### b. **tblClienti -> tblComenzi**

- Un client poate plasa mai multe comenzi
- Legătură: `tblComenzi.client_id → tblClienti.client_id`

#### c. **tblRestaurante -> tblComenzi**

- Un restaurant poate avea mai multe comenzi
- Legătură: `tblComenzi.restaurant_id → tblRestaurante.restaurant_id`

#### d. **tblRestaurante -> tblRezervari**

- Un restaurant poate avea mai multe rezervări
- Legătură: `tblRezervari.restaurant_id → tblRestaurante.restaurant_id`

### 2. Asociere N:M

→ O înregistrare din tabelul A poate fi asociată cu mai multe înregistrări din tabelul B și invers - o înregistrare din B poate fi asociată cu mai multe din A.

#### a. **tblComenzi -> tblProdus**

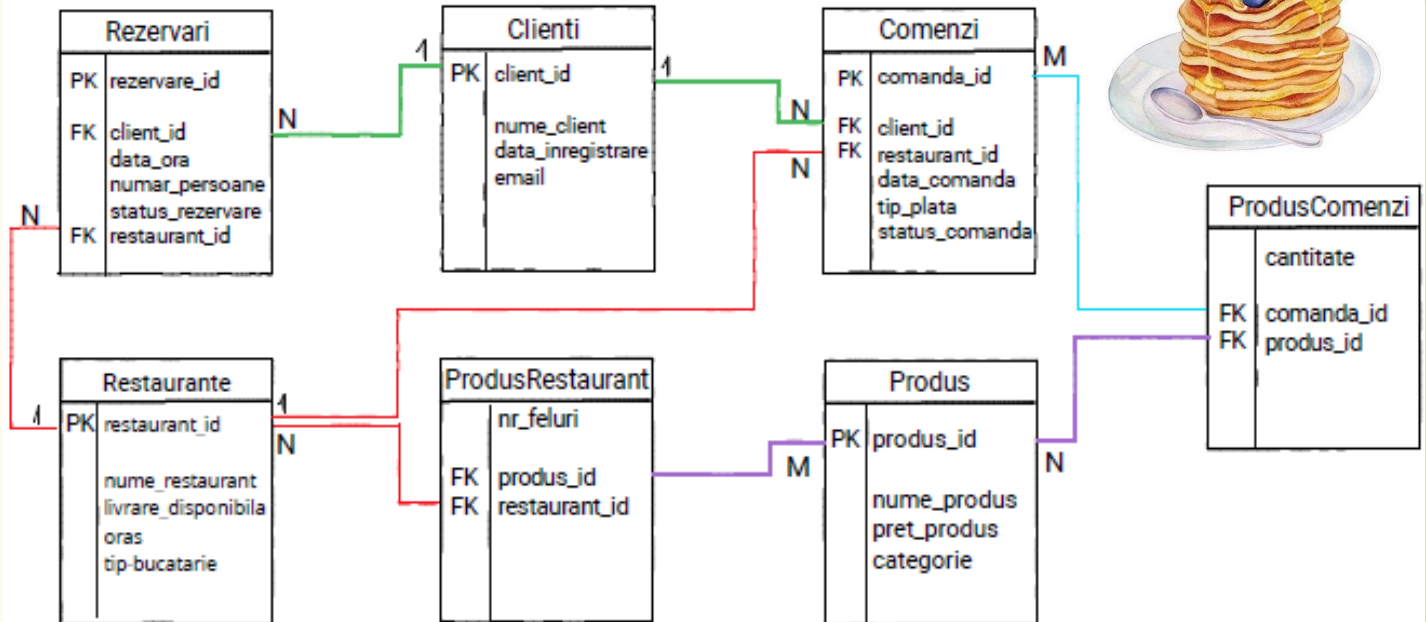
- Relație reală: **Comenzi → ProdusComenzi ← Produs**
- Conexiuni:
  - `tblProdusComenzi.produs_id`
  - `tblProdusComenzi.comanda_id`
- O comandă poate conține mai multe produse.
- Un produs poate fi parte din mai multe comenzi.

#### b. **tblProdus -> tblRestaurante**

- Relație reală: **Produs → ProdusRestaurant ← Restaurant**
- Conexiuni:
  - `tblProdusRestaurant.produs_id`
  - `tblProdusRestaurant.restaurant_id`
- Un produs poate fi la mai multe restaurante
- Un restaurant poate avea mai multe produse.



## Schemă relațională



## Model Entitate → Asociere

