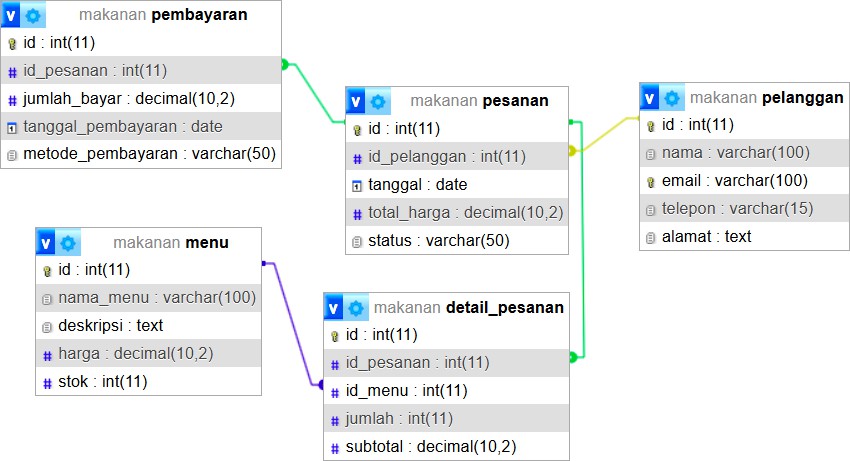
**Nama : Muhammad Abyan Agtasani**

**NIM : 22.01.53.0043**

**Nama Aplikasi : Sistem Kantin**

**Rancangan Database :**

****

# Tabel Pelanggan

Tabel ini menyimpan data pelanggan yang melakukan pemesanan makanan.

**Struktur Tabel:**

* + id\_pelanggan: ID unik untuk setiap pelanggan, bertipe INT dan menjadi **primary key**.
  + nama: Nama pelanggan, bertipe VARCHAR(100).
  + email: Alamat email pelanggan, bertipe VARCHAR(100) dan memiliki **unique constraint** agar tidak ada duplikasi email.
  + telepon: Nomor telepon pelanggan, bertipe VARCHAR(15).
  + alamat: Alamat pelanggan, bertipe TEXT.

**Relasi:**

* + Tidak ada relasi langsung dengan tabel lain, tetapi tabel ini memiliki relasi **one-to- many** dengan tabel Pesanan (satu pelanggan bisa membuat banyak pesanan).

# Tabel Menu

Tabel ini menyimpan data menu yang tersedia di restoran.

**Struktur Tabel:**

* + id\_menu: ID unik untuk setiap menu, bertipe INT dan menjadi **primary key**.
  + nama\_menu: Nama dari menu makanan, bertipe VARCHAR(100).
  + deskripsi: Deskripsi atau informasi tentang menu, bertipe TEXT.
  + harga: Harga per item menu, bertipe DECIMAL(10,2).
  + stok: Jumlah stok menu yang tersedia, bertipe INT.

**Relasi:**

* + Tidak ada relasi langsung dengan tabel lain, tetapi tabel ini memiliki relasi **many-to- many** dengan tabel Pesanan melalui tabel perantara Detail\_Pesanan. Artinya, satu menu bisa dipesan di banyak pesanan, dan satu pesanan bisa memiliki banyak menu.

# Tabel Pesanan

Tabel ini menyimpan data pesanan yang dilakukan oleh pelanggan.

**Struktur Tabel:**

* + id\_pesanan: ID unik untuk setiap pesanan, bertipe INT dan menjadi **primary key**.
  + id\_pelanggan: ID pelanggan yang melakukan pesanan, bertipe INT, dan menjadi

**foreign key** yang merujuk ke id\_pelanggan di tabel Pelanggan.

* + tanggal: Tanggal pesanan dilakukan, bertipe DATE.
  + total\_harga: Total harga dari pesanan tersebut, bertipe DECIMAL(10,2).
  + status: Status pesanan, bertipe VARCHAR(50) (misalnya: "Menunggu", "Diproses", "Selesai").

**Relasi:**

* + **One-to-many** dengan tabel Pelanggan: Setiap pelanggan bisa membuat banyak pesanan.
  + **One-to-many** dengan tabel Pembayaran: Setiap pesanan bisa memiliki satu pembayaran, tetapi satu pembayaran hanya untuk satu pesanan.
  + **Many-to-many** dengan tabel Menu melalui tabel Detail\_Pesanan: Pesanan bisa terdiri dari banyak menu.

# Tabel Detail\_Pesanan

Tabel ini menyimpan rincian menu yang dipesan dalam setiap pesanan.

**Struktur Tabel:**

* + id\_detail: ID unik untuk setiap detail pesanan, bertipe INT dan menjadi **primary key**.
  + id\_pesanan: ID pesanan, bertipe INT, dan menjadi **foreign key** yang merujuk ke

id\_pesanan di tabel Pesanan.

* + id\_menu: ID menu yang dipesan, bertipe INT, dan menjadi **foreign key** yang merujuk ke id\_menu di tabel Menu.
  + jumlah: Jumlah menu yang dipesan, bertipe INT.
  + subtotal: Subtotal harga untuk jumlah menu yang dipesan, bertipe DECIMAL(10,2).

**Relasi:**

* + **Many-to-one** dengan tabel Pesanan: Setiap detail pesanan berhubungan dengan satu pesanan.
  + **Many-to-one** dengan tabel Menu: Setiap detail pesanan berhubungan dengan satu menu.

# Tabel Pembayaran

Tabel ini menyimpan data pembayaran yang dilakukan untuk pesanan.

**Struktur Tabel:**

* + id\_pembayaran: ID unik untuk setiap pembayaran, bertipe INT dan menjadi **primary key**.
  + id\_pesanan: ID pesanan yang dibayar, bertipe INT, dan menjadi **foreign key** yang merujuk ke id\_pesanan di tabel Pesanan.
  + jumlah\_bayar: Jumlah yang dibayar untuk pesanan tersebut, bertipe DECIMAL(10,2).
  + tanggal\_pembayaran: Tanggal pembayaran dilakukan, bertipe DATE.
  + metode\_pembayaran: Metode pembayaran yang digunakan, bertipe VARCHAR(50)

(misalnya: "Tunai", "Transfer Bank").

**Relasi:**

* + **One-to-one** dengan tabel Pesanan: Setiap pembayaran hanya terkait dengan satu pesanan, dan satu pesanan hanya dapat memiliki satu pembayaran.

# Relasi Antar Tabel:

1. **Pelanggan ke Pesanan (One-to-many):**
   * Setiap pelanggan bisa memiliki banyak pesanan, tetapi setiap pesanan hanya dimiliki oleh satu pelanggan. Relasi ini menghubungkan id\_pelanggan di tabel Pesanan ke id\_pelanggan di tabel Pelanggan.
2. **Pesanan ke Detail\_Pesanan (One-to-many):**
   * Setiap pesanan bisa memiliki banyak rincian menu yang dipesan. Relasi ini menghubungkan id\_pesanan di tabel Detail\_Pesanan ke id\_pesanan di tabel Pesanan.
3. **Menu ke Detail\_Pesanan (One-to-many):**
   * Setiap menu bisa muncul di banyak pesanan. Relasi ini menghubungkan

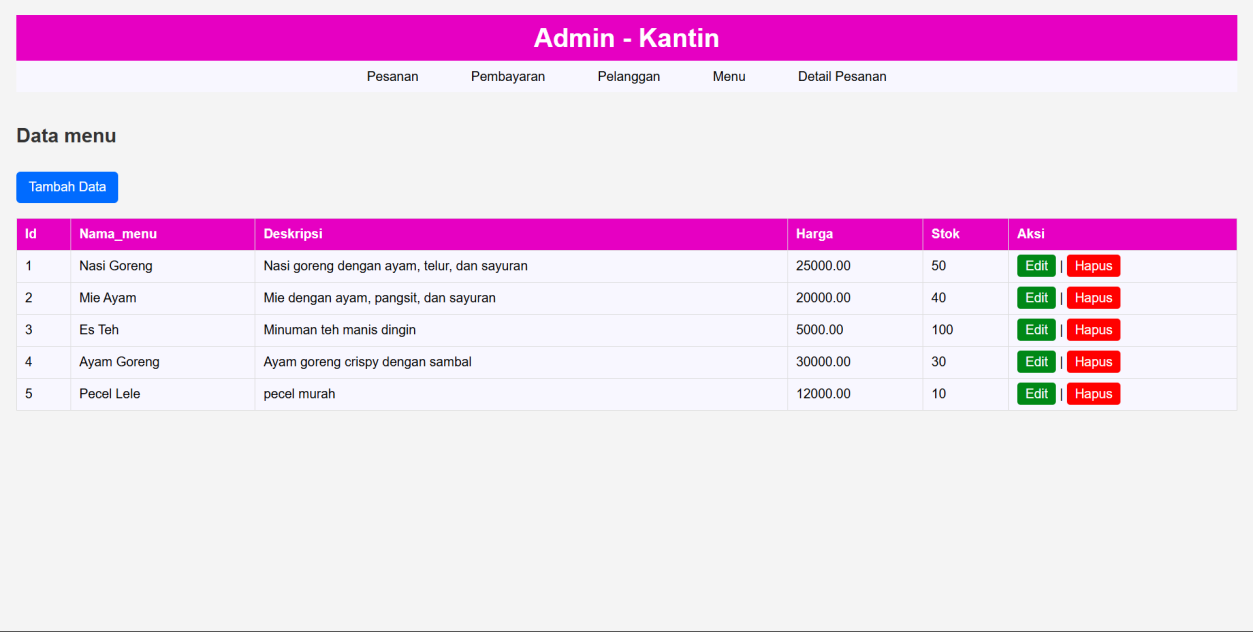
id\_menu di tabel Detail\_Pesanan ke id\_menu di tabel Menu.

1. **Pesanan ke Pembayaran (One-to-one):**
   * Setiap pesanan memiliki satu pembayaran. Relasi ini menghubungkan

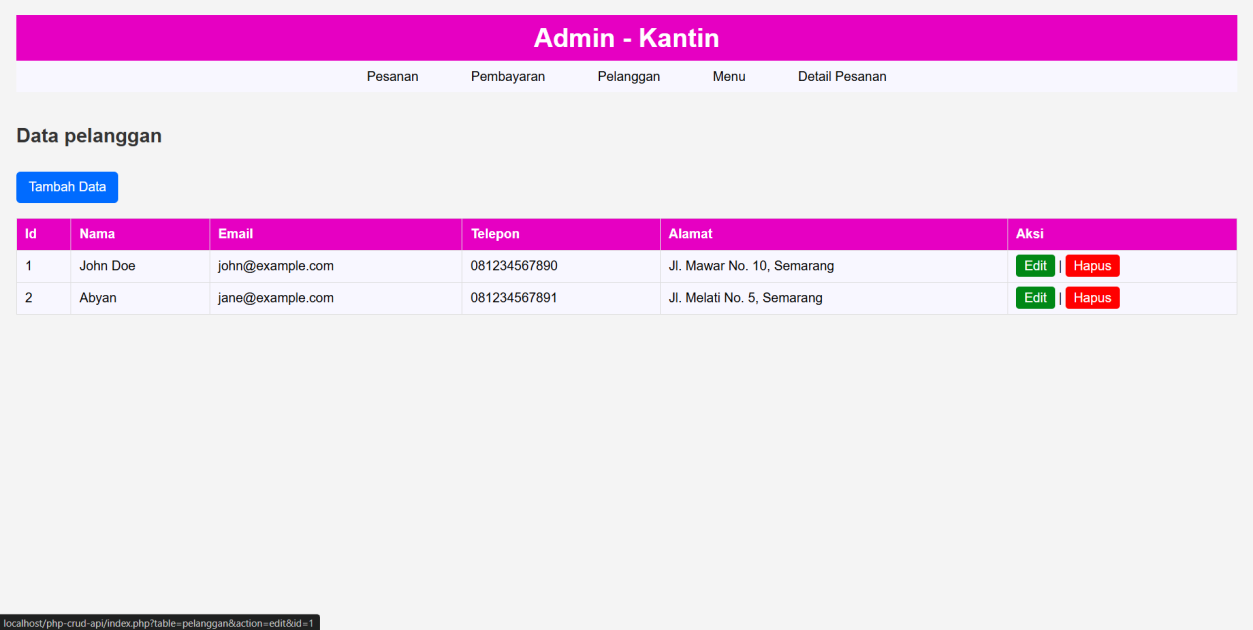
id\_pesanan di tabel Pembayaran ke id\_pesanan di tabel Pesanan.

1. **Menu ke Pesanan (Many-to-many melalui Detail\_Pesanan):**
   * Satu menu bisa dipesan dalam banyak pesanan, dan satu pesanan bisa mencakup banyak menu. Relasi ini diatur melalui tabel perantara Detail\_Pesanan.

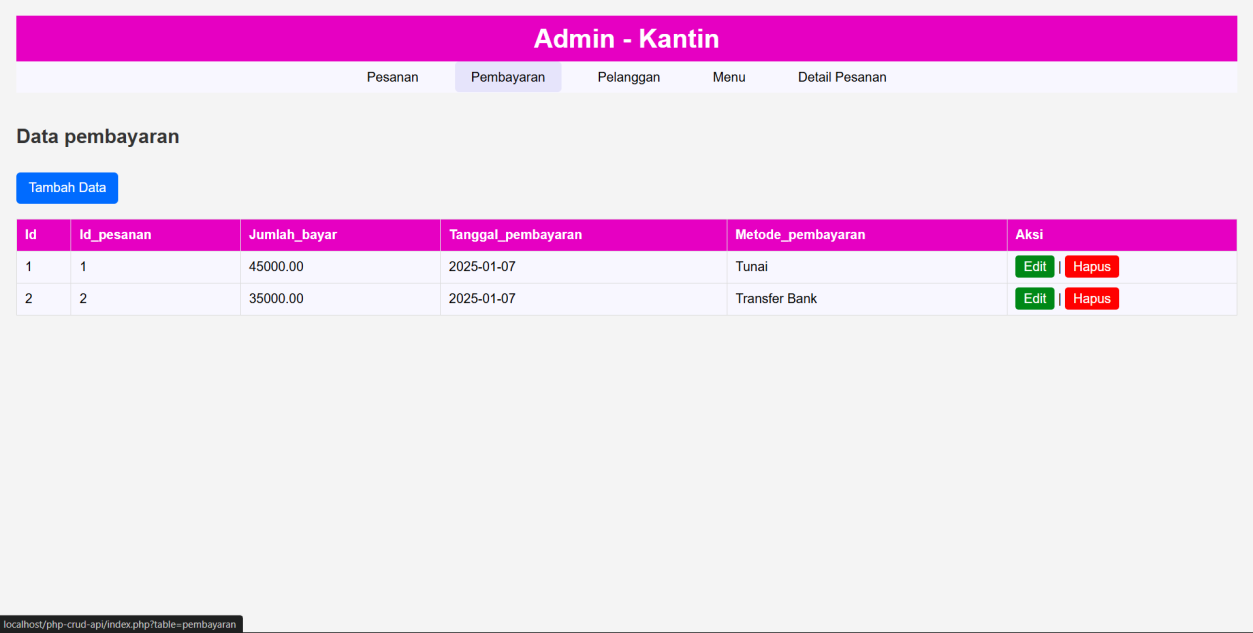
**Tampilan Tabel Menu**



**Tabel Pelanggan**

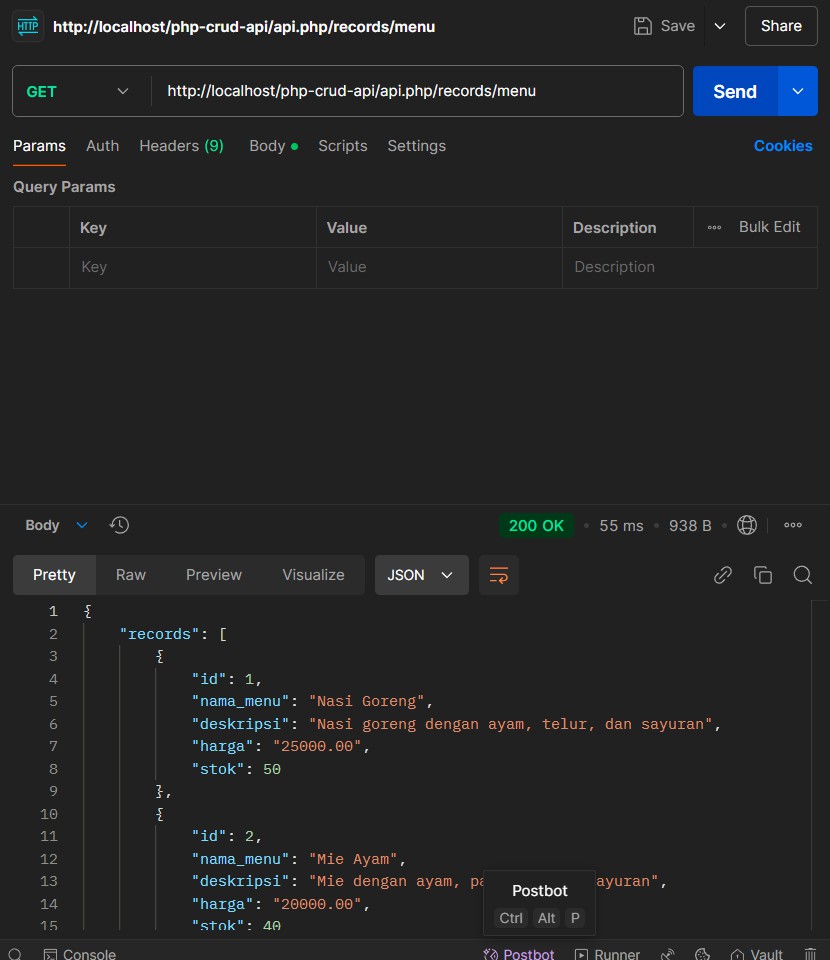
****

**Tabel Pembayaran**

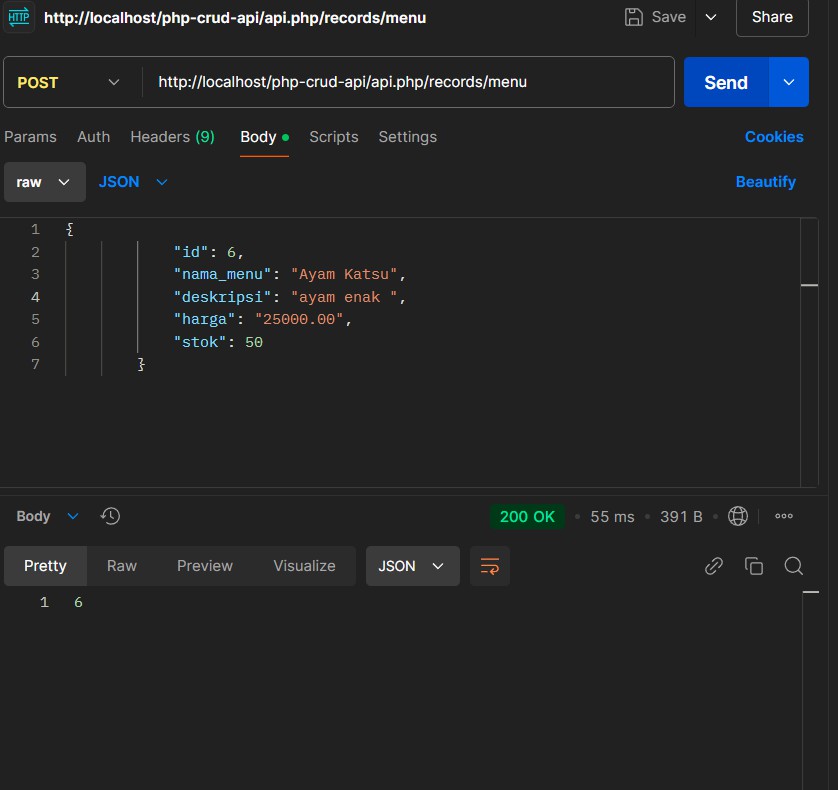
****

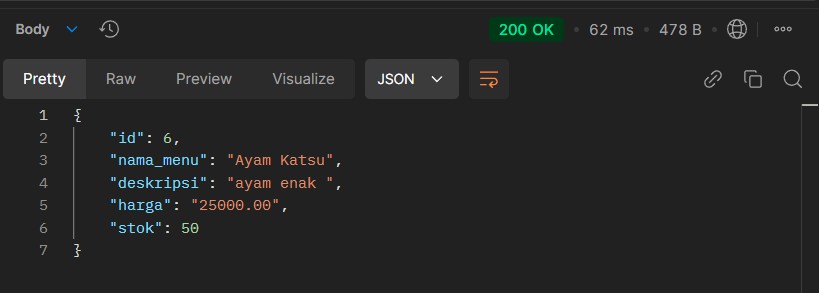
**METODE API**

**Tabel Menu GET**

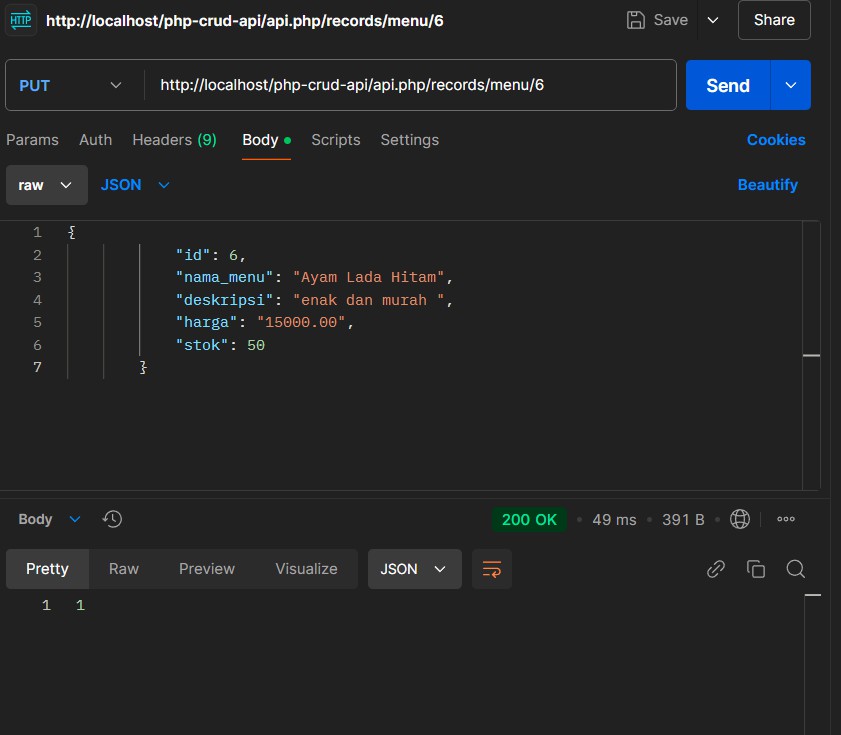


**POST**



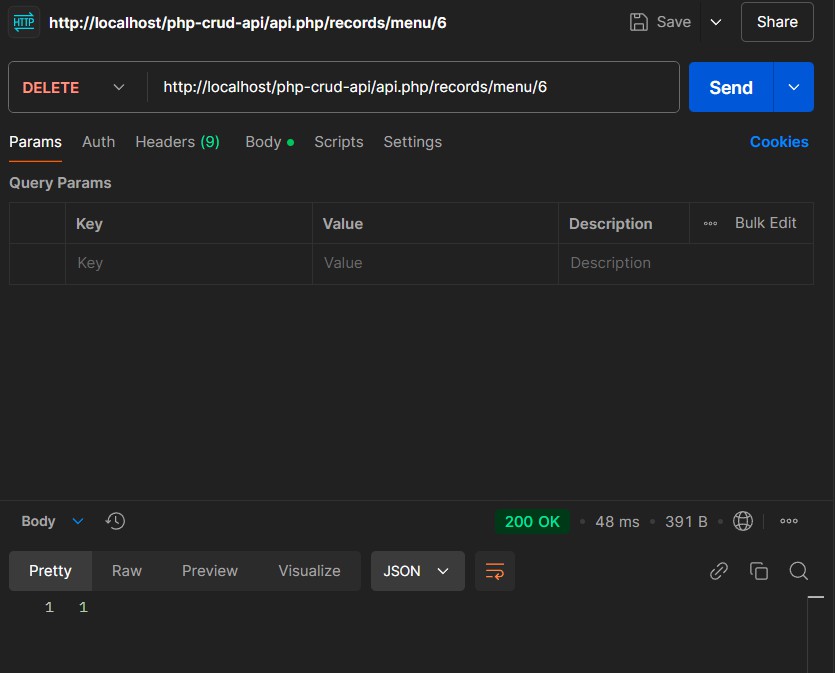
****

**PUT**



****

**DELETE**

****