

Nama : Bintang Arya Widi Syahputra

NIM : 22.01.53.0035

Ujian Tengah Semester

WEB Service

Rancangan tabel database website Pengelolaan Laundry

Tabel	Tindakan	Baris	Jenis	Penyortiran	Ukuran	Beban
<input type="checkbox"/> coupons	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/> discounts	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/> inventory	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_general_ci	16.0 KB	-
<input type="checkbox"/> laundry_items	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/> notifications	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/> payments	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/> reviews	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_general_ci	48.0 KB	-
<input type="checkbox"/> services	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	5	InnoDB	utf8mb4_general_ci	16.0 KB	-
<input type="checkbox"/> settings	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/> staff	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_general_ci	32.0 KB	-
<input type="checkbox"/> transactions	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	0	InnoDB	utf8mb4_general_ci	48.0 KB	-
<input type="checkbox"/> users	★ Jelajahi Struktur Cari Tambahkan Kosongkan Hapus	8	InnoDB	utf8mb4_general_ci	32.0 KB	-
12 tabel	Jumlah	13	InnoDB	utf8mb4_general_ci	384.0 KB	0 B

Membuat file dengan nama api.php :

```
<?php
header("Content-Type: application/json");
$host = 'localhost';
$db_name = 'laundry_db';
$username = 'root';
$password = '';

try {
    $conn = new PDO("mysql:host=$host;dbname=$db_name", $username, $password);
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
} catch (PDOException $e) {
    echo json_encode(['error' => $e->getMessage()]);
    exit();
}

$request_method = $_SERVER['REQUEST_METHOD'];

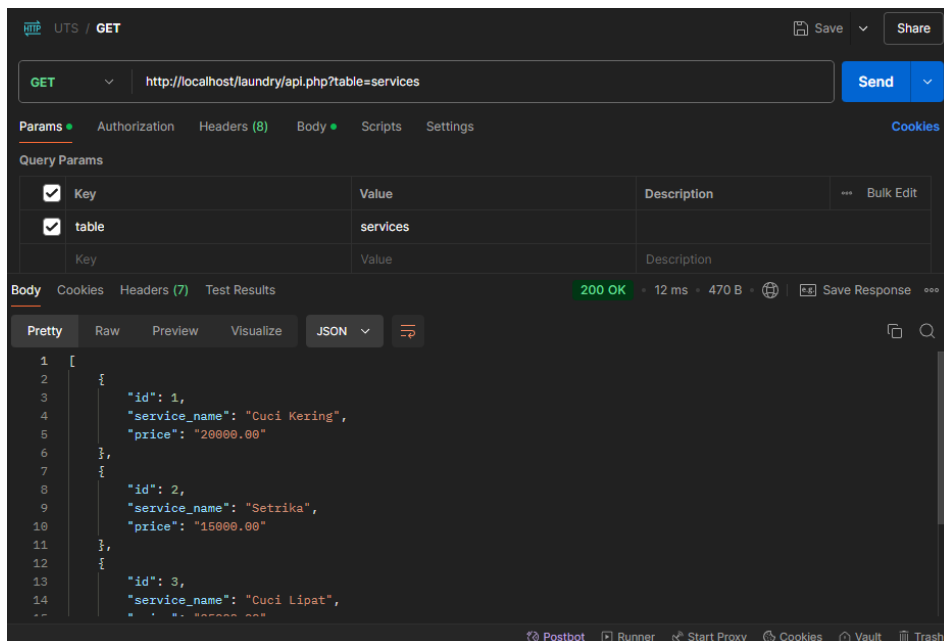
switch ($request_method) {
    case 'GET':
        if (!empty($_GET['table'])) {
            $table = $_GET['table'];
            $stmt = $conn->prepare("SELECT * FROM $table");
            $stmt->execute();
            $result = $stmt->fetchAll(PDO::FETCH_ASSOC);
            echo json_encode($result);
        } else {
            echo json_encode(['error' => 'Table name is required']);
        }
        break;

    case 'POST':
        if (!empty($_GET['table'])) {
            $table = $_GET['table'];
            $data = json_decode(file_get_contents('php://input'), true);
```

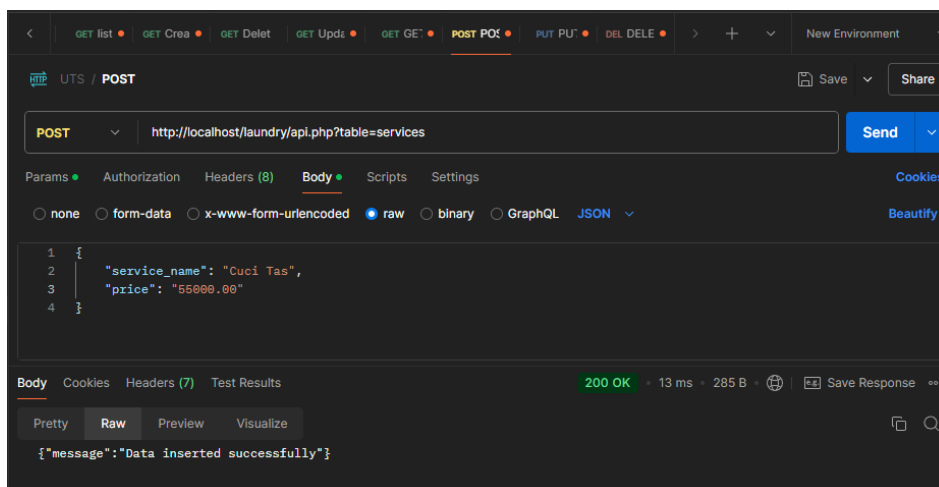
```
        case 'PUT':
            if (!empty($_GET['table']) && !empty($_GET['id'])) {
                $table = $_GET['table'];
                $id = $_GET['id'];
                $data = json_decode(file_get_contents('php://input'), true);
                $set = "";
                foreach ($data as $key => $value) {
                    $set .= "$key = $value, ";
                }
                $set = rtrim($set, ', ');
                $stmt = $conn->prepare("UPDATE $table SET $set WHERE id = :id");
                $data['id'] = $id;
                foreach ($data as $key => $value) {
                    $stmt->bindParam(":".$key, $value);
                }
                $stmt->execute();
                echo json_encode(['message' => 'Data updated successfully']);
            } else {
                echo json_encode(['error' => 'Table name and ID are required']);
            }
            break;

        case 'DELETE':
            if (!empty($_GET['table']) && !empty($_GET['id'])) {
                $table = $_GET['table'];
                $id = $_GET['id'];
                $stmt = $conn->prepare("DELETE FROM $table WHERE id = :id");
                $stmt->bindParam(":'id'", $id);
                $stmt->execute();
                echo json_encode(['message' => 'Data deleted successfully']);
            } else {
                echo json_encode(['error' => 'Table name and ID are required']);
            }
            break;
```

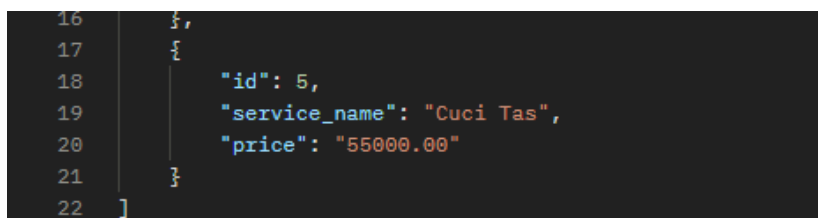
GET : Merupakan perintah api yang digunakan untuk menampilkan tabel services pada database laundry. GET digunakan untuk mengambil sumber daya dari server.



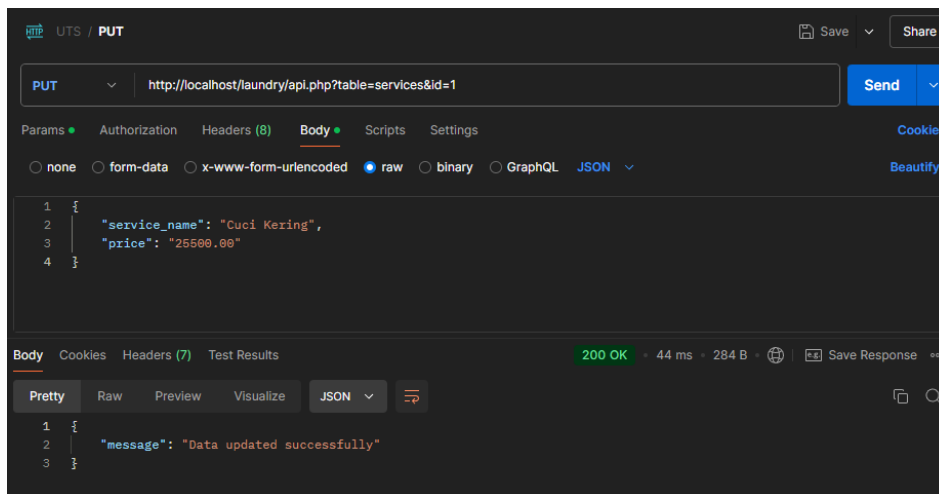
POST : Perintah API yang digunakan untuk menambahkan data baru ke dalam server tabel database yang dituju (Tabel services).



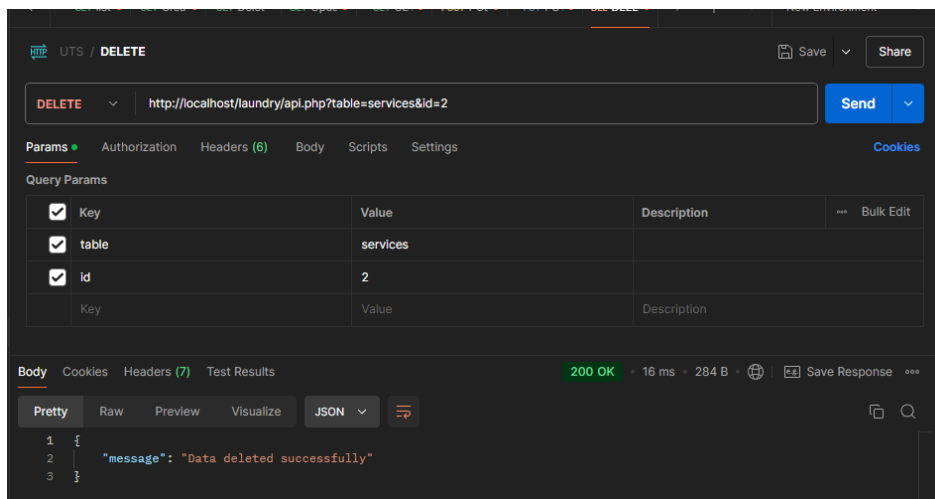
Hasil dari POST :



PUT : Merupakan perintah API yang dipakai untuk melakukan suatu update pada isi tabel database, dalam contoh berikut melakukan update harga pada services “Cuci Kering” dari 20000 menjadi 25500.



DELETE : Merupakan salah satu perintah API yang digunakan untuk menghapus data pada database, dalam kasus ini DELETE digunakan untuk menghapus data No. 2 pada tabel services.



Hasil dari DELETE :

