**Nama : Lintang putra Risaldi**

**Nim : 21.01.55.0013**

1. **Membuat data Base**

CREATE TABLE vehicles (

id INT AUTO\_INCREMENT PRIMARY KEY,

brand text(15) NOT NULL,

model text(15) NOT NULL,

year int(10) NOT NULL,

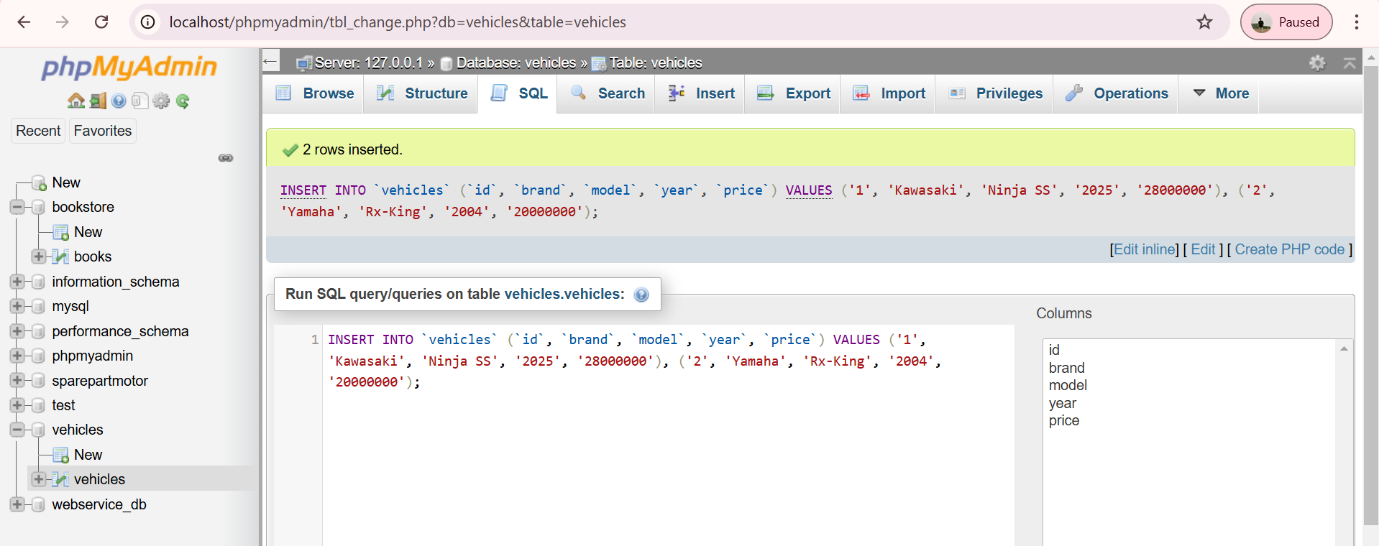
price INT NOT NULL

);

INSERT INTO vehicles (id,brand,model,year,price) VALUES

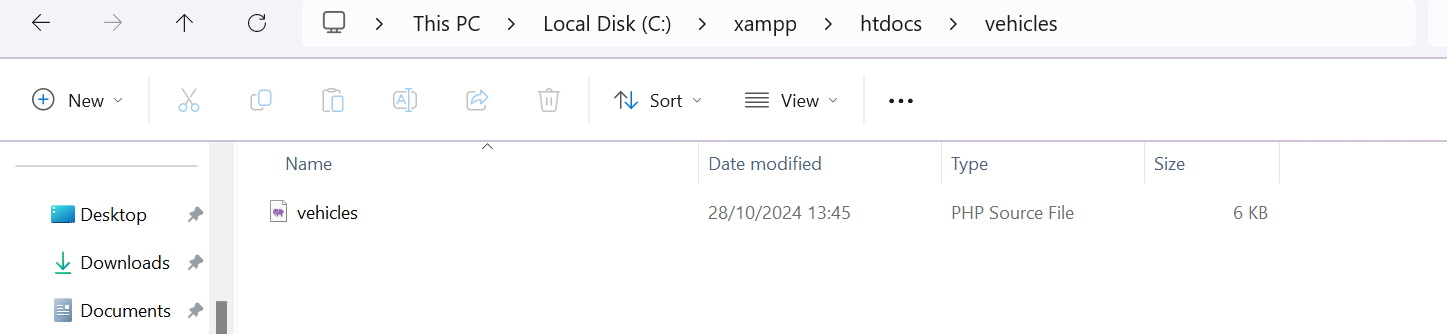
(‘1’,’Kawasaki’,’Ninja SS’,’2015’,’28000000’),

(‘2,Yamaha,’Rx-King’,’2004’,’20000000’);



1. **Membuat File Di File C**

Membuat folder di dalam C-Xampp-Htdock-vehicles.php



1. Program yang ada pada vehicles.php

<?php

header("Content-Type: application/json; charset=UTF-8");

header("Access-Control-Allow-Origin: \*");

header("Access-Control-Allow-Methods: GET, POST, PUT, DELETE");

header("Access-Control-Allow-Headers: Content-Type, Access-Control-Allow-Headers, modelization, X-Requested-With");

$method = $\_SERVER['REQUEST\_METHOD'];

$request = [];

if (isset($\_SERVER['PATH\_INFO'])) {

$request = explode('/', trim($\_SERVER['PATH\_INFO'],'/'));

}

function getConnection() {

$host = 'localhost';

$db = 'vehicles';

$user = 'root';

$pass = ''; // Ganti dengan password MySQL Anda jika ada

$charset = 'utf8mb4';

$dsn = "mysql:host=$host;dbname=$db;charset=$charset";

$options = [

PDO::ATTR\_ERRMODE => PDO::ERRMODE\_EXCEPTION,

PDO::ATTR\_DEFAULT\_FETCH\_MODE => PDO::FETCH\_ASSOC,

PDO::ATTR\_EMULATE\_PREPARES => false,

];

try {

return new PDO($dsn, $user, $pass, $options);

} catch (\PDOException $e) {

throw new \PDOException($e->getMessage(), (int)$e->getCode());

}

}

function response($status, $data = NULL) {

header("HTTP/1.1 " . $status);

if ($data) {

echo json\_encode($data);

}

exit();

}

$db = getConnection();

switch ($method) {

case 'GET':

if (!empty($request) && isset($request[0])) {

$id = $request[0];

$stmt = $db->prepare("SELECT \* FROM vehicles WHERE id = ?");

$stmt->execute([$id]);

$vehicles = $stmt->fetch();

if ($vehicles) {

response(200, $vehicles);

} else {

response(404, ["message" => "vehicles not found"]);

}

} else {

$stmt = $db->query("SELECT \* FROM vehicles");

$vehicles = $stmt->fetchAll();

response(200, $vehicles);

}

break;

case 'POST':

$data = json\_decode(file\_get\_contents("php://input"));

if (!isset($data->brand) || !isset($data->model) || !isset($data->year)) {

response(400, ["message" => "Missing required fields"]);

}

$sql = "INSERT INTO vehicles (brand, model, price, year) VALUES (?, ?, ?)";

$stmt = $db->prepare($sql);

if ($stmt->execute([$data->brand, $data->model, $data->price, $data->year])) {

response(201, ["message" => "vehicles created", "id" => $db->lastInsertId()]);

} else {

response(500, ["message" => "Failed to create vehicles"]);

}

break;

case 'PUT':

if (empty($request) || !isset($request[0])) {

response(400, ["message" => "vehicles ID is required"]);

}

$id = $request[0];

$data = json\_decode(file\_get\_contents("php://input"));

if (!isset($data->brand) || !isset($data->model) || !isset($data->year)) {

response(400, ["message" => "Missing required fields"]);

}

$sql = "UPDATE vehicles SET brand = ?, model = ?, price = ?, year = ? WHERE id = ?";

$stmt = $db->prepare($sql);

if ($stmt->execute([$data->brand, $data->model, $data->price, $data->year, $id])) {

response(200, ["message" => "vehicles updated"]);

} else {

response(500, ["message" => "Failed to update vehicles"]);

}

break;

case 'DELETE':

if (empty($request) || !isset($request[0])) {

response(400, ["message" => "vehicles ID is required"]);

}

$id = $request[0];

$sql = "DELETE FROM vehicles WHERE id = ?";

$stmt = $db->prepare($sql);

if ($stmt->execute([$id])) {

response(200, ["message" => "vehicles deleted"]);

} else {

response(500, ["message" => "Failed to delete vehicles"]);

}

break;

default:

response(405, ["message" => "Method not allowed"]);

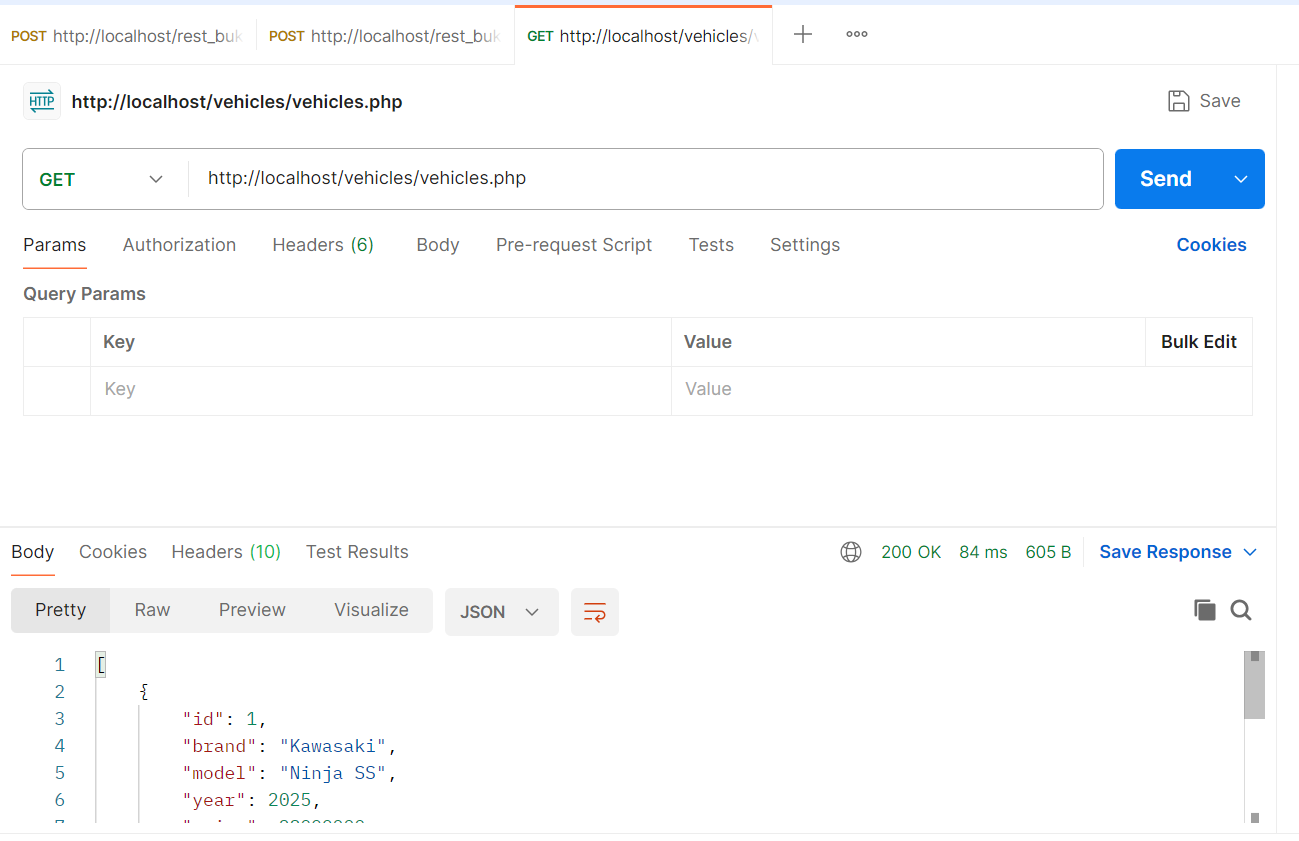
break;

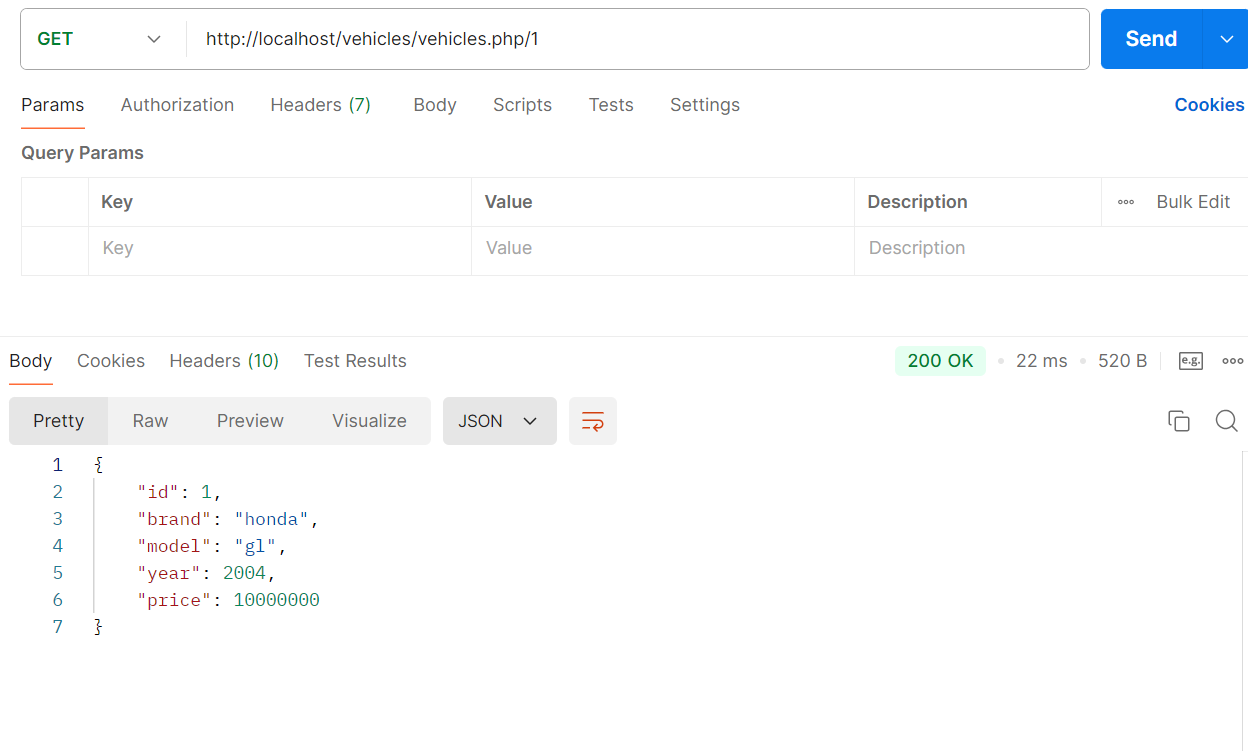
}

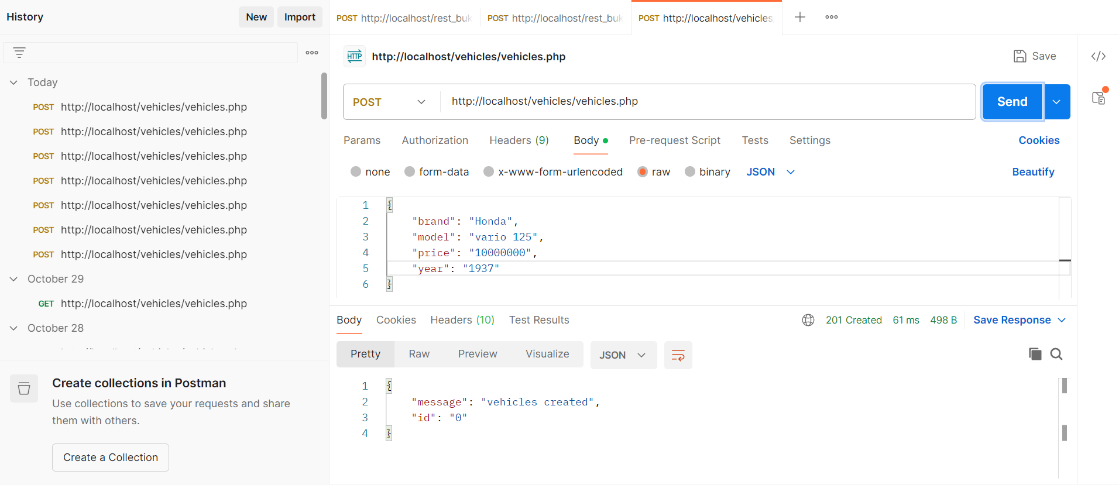
?>

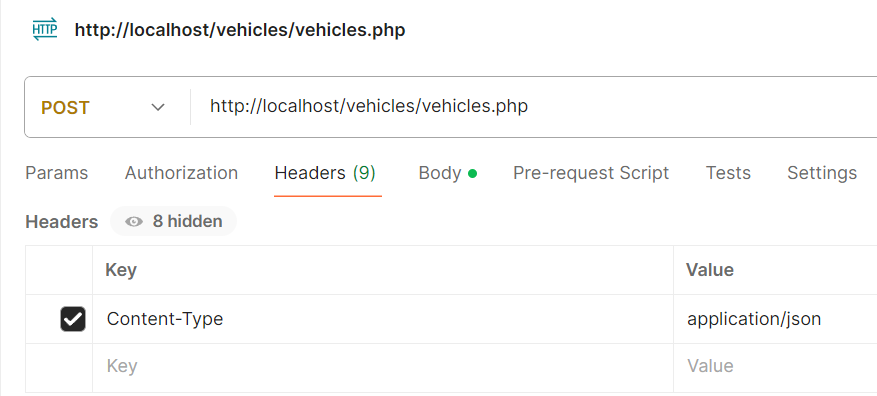
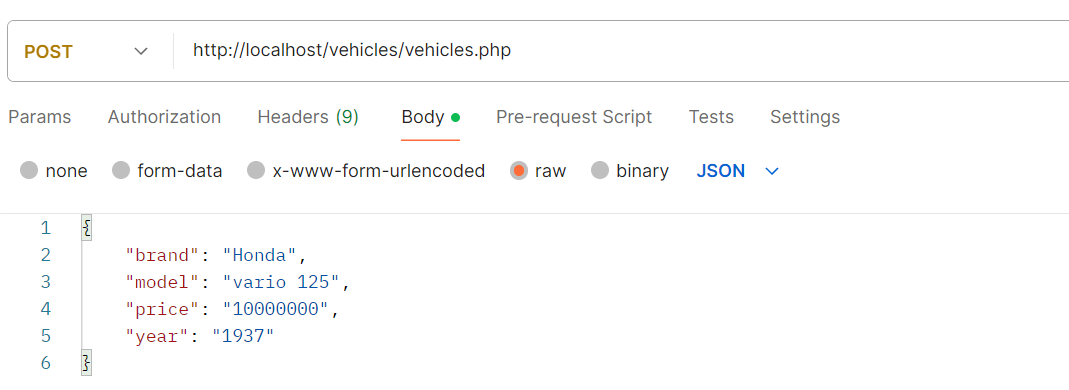
1. Hasil untuk menampilkan seluruh data

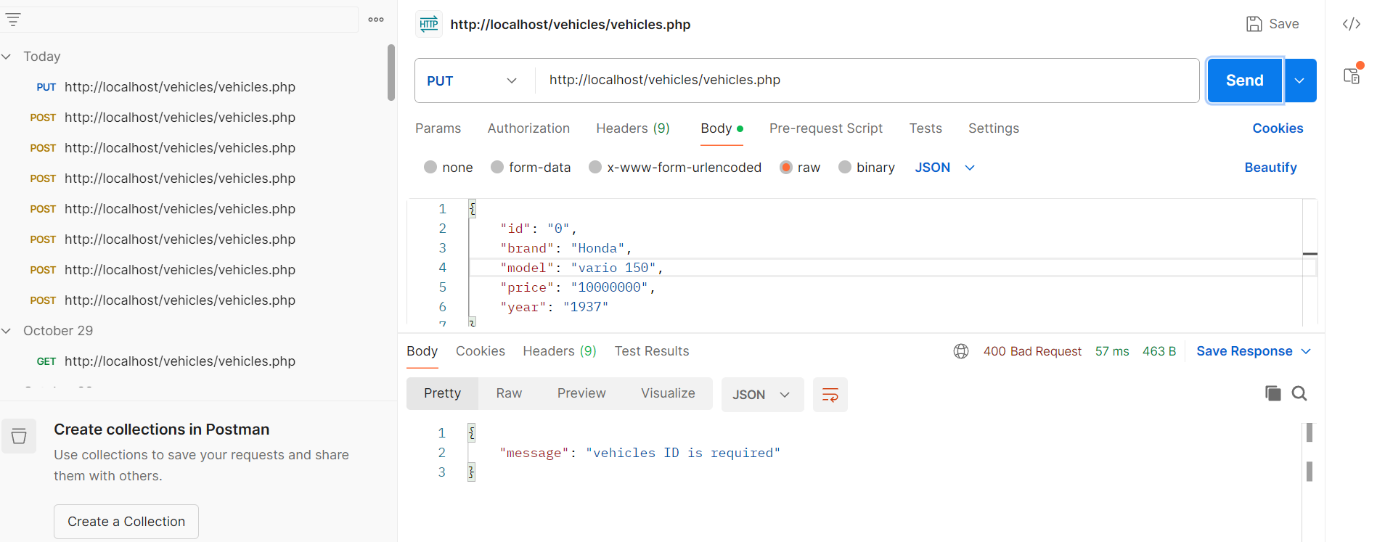
GET -> <http://localhost/vehicles/vehicles.php>

Di atas adalah hasil dari penampilan seluruh data pada mysql.



1. Post



1. Put
2. Delete

