

MODUL 06

RESTful API - DELETE

Materi

1. RESTful API

REST (*Representational State Transfer*) adalah arsitektur API (*Application Programming Interface*) yang menyediakan komunikasi *client-server* untuk aplikasi web melalui protokol HTTP. RESTful API merupakan implementasi dari REST API yang benar-benar mengikuti semua prinsip REST dengan baik. RESTful API beroperasi berdasarkan prinsip CRUD (*Create, Read, Update, Delete*).

2. DELETE

Salah satu prinsip CRUD yang digunakan RESTful API adalah fungsi *Delete* atau biasa dinamakan metode DELETE. DELETE pada RESTful API digunakan untuk menghapus data yang ada pada sistem.

Deskripsi Tugas

Pada praktikum ini, mahasiswa akan mempraktikkan pembuatan fungsi DELETE dalam RESTful API. Mahasiswa akan belajar memahami struktur kode program yang paling mendasar dari pembuatan RESTful API dengan fungsi DELETE.

Praktikum

| No. | Langkah-langkah |
|-----|--|
| 1. | <p>Pertama buat file dengan nama “post.php” di dalam folder “api” yang telah dibuat sebelumnya</p> <pre><?php header("Content-Type: application/json"); require "data.php"; header("Content-Type: application/json");</pre> <p>➤ Menetapkan header response agar output yang dikirimkan ke client berupa JSON.</p> |

| | |
|----|---|
| | <pre>require "data.php";</pre> <ul style="list-style-type: none"> ➤ Mengimpor file data.php, yang berisi fungsi getData() dan saveData(), digunakan untuk membaca dan menyimpan data dalam format JSON. |
| 2. | <pre>\$data = getData(); \$id = isset(\$_GET['id']) ? \$_GET['id'] : null;</pre> <pre>\$data = getData();</pre> <ul style="list-style-type: none"> ➤ Memanggil fungsi getData() untuk membaca isi file JSON dan menyimpannya dalam variabel \$data sebagai array. <pre>\$id = isset(\$_GET['id']) ? \$_GET['id'] : null;</pre> <ul style="list-style-type: none"> ➤ Mengecek apakah ada parameter id yang dikirim melalui query string (misalnya ?id=1). Jika id ada, nilainya disimpan ke dalam \$id, jika tidak, maka \$id akan bernilai null. |
| 3. | <pre>if (!\$id !isset(\$data[\$id])) { http_response_code(404); echo json_encode(["status" => 404, "error" => "Item Not Found"]); exit; }</pre> <p>Jika tidak ada id dalam URL atau id tidak ditemukan dalam data JSON maka API mengembalikan 404 Not Found.</p> |
| 4. | <pre>unset(\$data[\$id]); saveData(\$data);</pre> <pre>unset(\$data[\$id]);</pre> <ul style="list-style-type: none"> ➤ Menghapus item dari array berdasarkan id. <pre>saveData(\$data);</pre> <ul style="list-style-type: none"> ➤ Menyimpan perubahan ke file JSON. |
| 5. | <pre>http_response_code(200); echo json_encode(["status" => 200, "message" => "Item Deleted"]);</pre> <p>Jika berhasil, API mengembalikan 200 OK.</p> |