

SW-II

SISTEMAS WEB II

Prof. Anderson Vanin

AULA 08 – Continuação API PHP

Alterações na API

- **Geração automática de IDs únicos:** o código verifica o maior ID existente e atribui o próximo número sequencial.
- **Novo endpoint GET com parâmetro id:** permite buscar os dados de um usuário específico por `?id=valor`.

Alterações no código da aula anterior

api.php

```
1  <?php
2  //CABEÇALHO
3  header("Content-Type: application/json; charset=UTF-8"); //DEFINE O TIPO DE RESPOSTA
4
5  $metodo = $_SERVER['REQUEST_METHOD'];
6  //echo "Método da requisição: " . $metodo;
7
8  // RECUPERA O ARQUIVO JSON NA MESMA PASTA DO PROJETO
9  $arquivo = 'usuarios.json';
10
11  // VERIFICA SE O ARQUIVO EXISTE, SE NÃO EXISTIR CRIA UM COM ARRAY VAZIO
12  if (!file_exists($arquivo)) {
13      file_put_contents($arquivo, json_encode([], JSON_PRETTY_PRINT | JSON_UNESCAPED_UNICODE));
14  }
15
16  // LÊ O CONTEÚDO DO ARQUIVO JSON
17  $usuarios = json_decode(file_get_contents($arquivo), true);
18
```

Alterações no código da aula anterior

```
18
19     switch ($metodo) {
20         case 'GET':
21             // Verifica se há um parâmetro "id" na URL
22             if (isset($_GET['id'])) {
23                 $id = intval($_GET['id']);
24                 $usuario_encontrado = null;
25
26                 // Procura o usuário pelo ID
27                 foreach ($usuarios as $usuario) {
28                     if ($usuario['id'] == $id) {
29                         $usuario_encontrado = $usuario;
30                         break;
31                     }
32                 }
33
34                 if ($usuario_encontrado) {
35                     echo json_encode($usuario_encontrado, JSON_PRETTY_PRINT | JSON_UNESCAPED_UNICODE);
36                 } else {
37                     http_response_code(404);
38                     echo json_encode(["erro" => "Usuário não encontrado."], JSON_UNESCAPED_UNICODE);
39                 }
40             } else {
41                 // Retorna todos os usuários
42                 echo json_encode($usuarios, JSON_PRETTY_PRINT | JSON_UNESCAPED_UNICODE);
43             }
44         break;
```

Alterações no código da aula anterior

```
45 case 'POST':
46     $dados = json_decode(file_get_contents('php://input'), true);
47
48     // VERIFICA CAMPOS OBRIGATÓRIOS (sem exigir o ID)
49     if (!isset($dados["nome"]) || !isset($dados["email"])) {
50         http_response_code(400);
51         echo json_encode(["erro" => "Nome e email são obrigatórios."], JSON_UNESCAPED_UNICODE);
52         exit;
53     }
54     // GERA UM NOVO ID ÚNICO
55     $novo_id = 1;
56     if (!empty($usuarios)) {
57         $ids = array_column($usuarios, 'id');
58         $novo_id = max($ids) + 1;
59     }
60
61     $novo_usuario = [
62         "id" => $novo_id,
63         "nome" => $dados["nome"],
64         "email" => $dados["email"],
65     ];
66     // ADICIONA O NOVO USUÁRIO
67     $usuarios[] = $novo_usuario;
68
69     // SALVA NO ARQUIVO
70     file_put_contents($arquivo, json_encode($usuarios, JSON_PRETTY_PRINT | JSON_UNESCAPED_UNICODE));
71
72     // RETORNA CONFIRMAÇÃO
73     echo json_encode([
74         "mensagem" => "Usuário inserido com sucesso!",
75         "usuario" => $novo_usuario
76     ], JSON_UNESCAPED_UNICODE);
77     break;
```

Alterações no código da aula anterior

```
78         default:
79             // echo "MÉTODO NÃO ENCONTRADO!";
80             // break;
81             http_response_code(405); // Método não permitido
82             echo json_encode(["erro" => "Método não permitido!"], JSON_UNESCAPED_UNICODE);
83             break;
84     }
85     ?>
```

Exemplos de uso

- Listar todos os usuários

GET

Status: 200 OK Size: 193 Bytes Time: 14 ms

Response Headers 6 Cookies Results Docs

```
1  [  
2    {  
3      "id": 1,  
4      "nome": "Maria Souza",  
5      "email": "maria@email.com"  
6    },  
7    {  
8      "id": 2,  
9      "nome": "João Silva",  
10     "email": "joao@email.com"  
11   }  
12 ]
```

- Buscar usuário específico (ID 3, por exemplo)

GET

Query Headers 2 Auth Body Tests Pre Run

Query Parameters

<input checked="" type="checkbox"/>	id	2
<input type="checkbox"/>	parameter	value

Status: 200 OK Size: 73 Bytes Time: 4 ms

Response Headers 6 Cookies Results Docs

```
1  {  
2    "id": 2,  
3    "nome": "João Silva",  
4    "email": "joao@email.com"  
5  }
```

Exemplos de uso

- Inserir novo usuário (sem enviar ID)

The screenshot displays a REST client interface with the following details:

- Request:**
 - Method: POST
 - URL: `http://localhost/aula06_swii/api.php`
 - Tab: Body¹
 - Format: JSON
 - Content:

```
1 {
2   "nome": "Fulano da Silva",
3   "email": "fulano@email.com"
4 }
```
- Response:**
 - Status: 200 OK
 - Size: 116 Bytes
 - Time: 5 ms
 - Tab: Response
 - Content:

```
1 {
2   "mensagem": "Usuário inserido com sucesso!",
3   "usuario": {
4     "id": 4,
5     "nome": "Fulano da Silva",
6     "email": "fulano@email.com"
7   }
8 }
```