

Primeiro passo: Adicionar o campo no script que cria o banco de dados (criabanco.php)

```
34 // SQL para criar a tabela 'cliente'
35 $sql = "CREATE TABLE cliente (
36     idcli INT(11) AUTO_INCREMENT PRIMARY KEY,
37     nome VARCHAR(255) NOT NULL,
38     categoria VARCHAR(255) NOT NULL,
39     email VARCHAR(50) NOT NULL
40     * adicionar o novo campo, não esquecer de
      adicionar uma vírgula antes
      );
```

Segundo passo: (testeApi.html) inserir um input para o novo campo:

```
Id <input type="text" id="id" name="id" value="0" size="3" readonly /><br /><br />
Nome <input type="text" id="nome" name="nome" value="" size="35" /><br /><br />
Categoria <input type="text" id="categoria" name="categoria" value="" size="20" /><br /><br />
E-mail<input type="text" id="email" name="email" value="" /> <p></p>
```

adicionar aqui o novo input, para o novo campo que foi criado no banco de dados
`campo <input type="text" id="campo" name="campo" value="" /> <p></p>`

Terceiro passo: Adicionar a referência do campo no envio da informação para a API

```
async function post() {
    try {
        const response = await fetch("http://localhost/api/testeApi.php/cliente", {
            method: "POST",
            headers: {
                "Content-Type": "application/json",
            },
            body: JSON.stringify({
                nome: document.getElementById("nome").value,
                categoria: document.getElementById("categoria").value,
                email: document.getElementById("email").value,
                campo: document.getElementById("campo").value
                * adicionar a referência ao campo criado no banco de dados
            })
        });
    }
}
```

Quarto passo: receber as informações enviadas pelo frontend para a API:

```

case 'POST':
    // Insere um novo cliente no banco de dados
    $data = json_decode(file_get_contents("php://input"), true);
    $nome = $data['nome']; // Assume que os dados são enviados em JSON
    $categoria = $data['categoria'];
    $email = $data['email'];
    $campo = $data["campo"];
    * recebendo informações do frontend

```

Quinto passo: inserir a referência do campo criado no banco de dados na instrução

SQL:

```

$sql = "INSERT INTO cliente (nome, categoria, email) VALUES ('$nome', '$categoria', '$email')";

```



- ⚠ Atenção, deve ser inserido, logo após o último campo, adicionando um virgula para então adicionar o campo.
- ⚠ A primeira inserção é o nome do campo e a segunda a variável que contém o valor do campo (por padrão utilizar o mesmo nome)

Sexto passo: Retornar as informações para a página que chamou a api
(endPoint)

```

// Retorna a resposta em formato JSON
header('Content-Type: application/json');
echo json_encode($response);

```

- ⚠ Json_encode codifica a informação gerando uma string JSON que é em seguida retornada ao frontend.

Observação:

msg é o atributo que contém a resposta ao frontend:

```

if ($conn->query($sql) === TRUE) {
    if ($conn->affected_rows > 0) {
        $response['message'] = "Cliente adicionado com sucesso!";
    } else {
        $response['message'] = "Erro ao adicionar cliente: " . $conn->error;
    }
}

```

■ No exemplo do cadastro, veja que `$response['message']` armazena a resposta ao servidor. Se a execução da sql pelo método `query` da conexão (`conn`) for verdadeira então “Cliente adicionado com sucesso”, senão “erro ao adicionar cliente”.