

# CRUD com PHP e MySQL

## Criar (CREATE)

Para inserir um recurso no banco dados usando PHP, precisamos coletar os dados do usuário. Uma forma eficiente é usar um formulário HTML, usando o método POST para enviar os dados e o atributo `action` para especificar o script PHP que processará os dados.

Exemplo de formulário HTML para cadastrar clientes:

```
<form method="POST" action="cadastrar_cliente.php">
  <label for="nome">Nome:</label>
  <input type="text" id="nome" name="nome" required>

  <label for="email">Email:</label>
  <input type="email" id="email" name="email" required>

  <label for="endereco">Endereço:</label>
  <input type="text" id="endereco" name="endereco">

  <label for="data_nasc">Data de Nascimento:</label>
  <input type="date" id="data_nasc" name="data_nasc">

  <button type="submit">Cadastrar</button>
</form>
```

Após o envio do formulário, o script `cadastrar_cliente.php` processará os dados.

```
<?php
// cadastrar_cliente.php
require_once('conexao.php');
if ($_SERVER["REQUEST_METHOD"] == "POST") {

    // Obter dados do formulário
    $nome = $_POST['nome'];
    $email = $_POST['email'];
    $endereco = $_POST['endereco'];
    $data_nasc = $_POST['data_nasc'];

    $sql = "INSERT INTO clientes (nome, email, endereco, data_nasc) VALUES ('$nome', '$email',

    if (mysqli_query($conn, $sql)) {
        echo "Cliente cadastrado com sucesso!";
    } else {
        echo "Erro: " . mysqli_error($conn);
    }

    // Fechar conexão
    mysqli_close($conn);
}
```

## Ler (READ)

Para ler os dados do banco de dados, você pode usar uma consulta SQL `SELECT` e exibir os resultados em uma tabela HTML. Neste exemplo, estou usando um arquivo de template com a base HTML e o CSS necessário para estilizar a tabela.

```
<?php
require_once('conexao.php');

// Lista os clientes existentes
$sql = "SELECT * FROM clientes";
$result = mysqli_query($conn, $sql);

if (!$result) die("Erro na consulta: " . mysqli_error($conn));

// Exibe os clientes em tabela HTML
$title = "Lista de Clientes";
$content = "<table>
    <tr>
        <th>ID</th>
        <th>Nome</th>
        <th>Email</th>
        <th>Endereço</th>
        <th>Data de Nascimento</th>
    </tr>";

while ($row = mysqli_fetch_assoc($result)) {
    $content .= "<tr>
        <td>{$row['id']}</td>
        <td>{$row['nome']}</td>
        <td>{$row['email']}</td>
        <td>{$row['endereço']}</td>
        <td>{$row['data_nasc']}</td>
    </tr>";
}

$content .= "</table>";

// Carrega o template HTML
require_once('template.php');

// Fecha a conexão com o banco de dados
mysqli_close($conn);
```

# Atualizar (UPDATE)

Para atualizar os dados de um cliente, você pode usar uma consulta SQL `UPDATE` . Veja um exemplo em PHP:

```
<?php
// atualizar_cliente.php
require_once('conexao.php');

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    // Obter dados do formulário
    $id = $_POST['id'];
    $nome = $_POST['nome'];
    $email = $_POST['email'];
    $endereco = $_POST['endereco'];
    $data_nasc = $_POST['data_nasc'];

    $sql = "UPDATE clientes SET nome='$nome', email='$email', endereco='$endereco', data_nasc='$data_nasc'";

    if (mysqli_query($conn, $sql)) {
        echo "Cliente atualizado com sucesso!";
    } else {
        echo "Erro: " . mysqli_error($conn);
    }

    // Fechar conexão
    mysqli_close($conn);
}
```

Será necessário uma página para editar os dados do cliente. Você pode criar um formulário semelhante ao de cadastro, mas preenchendo os campos com os dados atuais do cliente. Quando o formulário for enviado, ele deve chamar o script `atualizar_cliente.php` para processar a atualização. Sabendo o ID do cliente, você pode buscar os dados atuais no banco de dados e preenchê-los no formulário.

```

<?php
// editar_cliente.php
require_once('conexao.php');

if (isset($_GET['id'])) {
    $id = $_GET['id'];
    $sql = "SELECT * FROM clientes WHERE id='$id'";
    $result = mysqli_query($conn, $sql);
    $cliente = mysqli_fetch_assoc($result);
}

?>

<html>
<head>
    <title>Editar Cliente</title>
</head>
<body>
    <h1>Editar Cliente</h1>
    <form method="POST" action="atualizar_cliente.php">
        <input type="hidden" name="id" value="<?php echo $cliente['id']; ?>">
        <label for="nome">Nome:</label>
        <input type="text" id="nome" name="nome" value="<?php echo $cliente['nome']; ?>" required>

        <label for="email">Email:</label>
        <input type="email" id="email" name="email" value="<?php echo $cliente['email']; ?>" required>

        <label for="endereco">Endereço:</label>
        <input type="text" id="endereco" name="endereco" value="<?php echo $cliente['endereco']; ?>">

        <label for="data_nasc">Data de Nascimento:</label>
        <input type="date" id="data_nasc" name="data_nasc" value="<?php echo $cliente['data_nasc']; ?>">

        <button type="submit">Atualizar</button>
    </form>
</body>
</html>

```

Na lista de clientes, você pode adicionar um link para editar cada cliente. Basicamente, mandamos via URL (Query String) o ID do cliente que queremos editar.

```

while ($row = mysqli_fetch_assoc($result)) {
    $content .= "<tr>
        <td>{$row['id']}

```

## Deletar (DELETE)

Para deletar um cliente, você pode usar uma consulta SQL `DELETE`. Veja um exemplo em PHP:

```

<?php
// deletar_cliente.php
require_once('conexao.php');

if (isset($_GET['id'])) {
    $id = $_GET['id'];
    $sql = "DELETE FROM clientes WHERE id='$id'";

    if (mysqli_query($conn, $sql)) {
        echo "Cliente deletado com sucesso!";
    } else {
        echo "Erro: " . mysqli_error($conn);
    }

    // Fechar conexão
    mysqli_close($conn);
}

```