

# CRUD com Bootstrap, PHP & MySQL – Parte IV

Esta é uma série de tutoriais, na qual você vai aprender uma forma eficaz de implementar um sistema de CRUD completo, usando o Bootstrap no front-end e PHP com MySQL no back end.

Agora, neste tutorial, você vai ver como criar os arquivos funções para implementar os formulários de inserção e a edição no cadastro de clientes. Além disso, vamos criar as funções de persistência dessas informações no banco de dados.

**CRUD** **Clientes**

### Novo Cliente

Nome / Razão Social		CNPJ / CPF		Data de Nascimento	
<input type="text"/>		<input type="text"/>		<input type="text"/>	
Endereço		Bairro	CEP	Data de Cadastro	
<input type="text"/>		<input type="text"/>	<input type="text"/>	<input type="text"/>	
Município	Telefone	Celular	UF	Inscrição Estadual	UF
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<input type="button" value="Salvar"/> <input type="button" value="Cancelar"/>					

©2016 - Web Dev Academy

Ao final deste tutorial, você vai ter uma estrutura básica de cadastro de clientes para reaproveitar no projeto, na qual o usuário vai poder preencher os dados do cadastro de clientes.

Aviso: Devido a outros projetos, vou adicionar as explicações de cada código aos poucos, durante a semana.



passou pelas partes anteriores desta série.

Se você ainda não estiver com o ambiente pronto, é só ver a [introdução desta série](#) para instalar os pré-requisitos, e criar o BD.

Este tutorial é um pouco extenso, e tem alguns códigos mais complexos. Separe um tempo para entender bem, e caso não entenda alguma parte, é só perguntar nos comentários.

## Passo 1: Crie a Função de Cadastro

Voltando à pasta “*customers*”, adicione estas a função de cadastro no arquivo `functions.php`:

```
<?php
/**
 * Cadastro de Clientes
 */
function add() {

    if (!empty($_POST['customer'])) {

        $today =
            date_create('now', new DateTimeZone('America/Sao_Paulo'));

        $customer = $_POST['customer'];
        $customer['modified'] = $customer['created'] = $today->format("Y-m-d H:i:s");

        save('customers', $customer);
        header('location: index.php');
    }
}
```

functions.php hosted with ❤ by GitHub

[view raw](#)

## Passo 2: Crie o Formulário de Cadastro

Agora, crie um arquivo chamado `add.php` na pasta `customers`. Esse arquivo vai ter a marcação do formulário de cadastro do cliente:

```
1 <?php
2     require_once('functions.php');
3     add();
4 ?>
5
6 <?php include(HEADER_TEMPLATE); ?>
```



```

<!-- area de campos do form -->
<hr />
<div class="row">
  <div class="form-group col-md-7">
    <label for="name">Nome / Razão Social</label>
    <input type="text" class="form-control" name="customer['name']">
  </div>

  <div class="form-group col-md-3">
    <label for="campo2">CNPJ / CPF</label>
    <input type="text" class="form-control" name="customer['cpf_cnpj']">
  </div>

  <div class="form-group col-md-2">
    <label for="campo3">Data de Nascimento</label>
    <input type="text" class="form-control" name="customer['birthdate']">
  </div>
</div>

<div class="row">
  <div class="form-group col-md-5">
    <label for="campo1">Endereço</label>
    <input type="text" class="form-control" name="customer['address']">
  </div>

  <div class="form-group col-md-3">
    <label for="campo2">Bairro</label>
    <input type="text" class="form-control" name="customer['hood']">
  </div>

  <div class="form-group col-md-2">
    <label for="campo3">CEP</label>
    <input type="text" class="form-control" name="customer['zip_code']">
  </div>

  <div class="form-group col-md-2">
    <label for="campo3">Data de Cadastro</label>
    <input type="text" class="form-control" name="customer['created']" disabled>
  </div>
</div>

<div class="row">
  <div class="form-group col-md-3">
    <label for="campo1">Município</label>
    <input type="text" class="form-control" name="customer['city']">
  </div>

```



```

</div>

<div class="form-group col-md-2">
  <label for="campo3">Celular</label>
  <input type="text" class="form-control" name="customer['mobile']">
</div>

<div class="form-group col-md-1">
  <label for="campo3">UF</label>
  <input type="text" class="form-control" name="customer['state']">
</div>

<div class="form-group col-md-2">
  <label for="campo3">Inscrição Estadual</label>
  <input type="text" class="form-control" name="customer['ie']">
</div>

<div class="form-group col-md-2">
  <label for="campo3">UF</label>
  <input type="text" class="form-control">
</div>
</div>

<div id="actions" class="row">
  <div class="col-md-12">
    <button type="submit" class="btn btn-primary">Salvar</button>
    <a href="index.php" class="btn btn-default">Cancelar</a>
  </div>
</div>
</form>

<?php include(FOOTER_TEMPLATE); ?>

```

add.php hosted with ❤ by GitHub

[view raw](#)

## Passo 3: Crie a Função de Persistência no BD

No arquivo database.php, implemente a função que vai inserir um registro no banco de dados:

```

1 <?php
2
3 /**
4  * Insere um registro no BD
5  */
6 function save($table = null, $data = null) {

```



```

118-120 $values = null;
121-123
124-126 //print_r($data);
127-129
130-132 foreach ($data as $key => $value) {
133-135     $columns .= trim($key, "'") . ",";
136-138     $values .= "'$value',";
139-141 }
142-144
145-147 // remove a ultima virgula
148-150 $columns = rtrim($columns, ',');
151-153 $values = rtrim($values, ',');
154-156
157-159 $sql = "INSERT INTO " . $table . "($columns)" . " VALUES " . "($values)";
160-162
163-165 try {
166-168     $database->query($sql);
169-171
172-174     $_SESSION['message'] = 'Registro cadastrado com sucesso.';
175-177     $_SESSION['type'] = 'success';
178-180
181-183 } catch (Exception $e) {
184-186
187-189     $_SESSION['message'] = 'Nao foi possivel realizar a operacao.';
190-192     $_SESSION['type'] = 'danger';
193-195 }
196-198
199-201 close_database($database);
202-204 }

```

database.php hosted with ❤ by GitHub

[view raw](#)

## Passo 4: Crie a Função de Edição/Atualização

Agora, vamos fazer a parte de edição, ou atualização dos clientes.

O primeiro passo é implementar a função *edit()* no módulo de clientes, ou seja, no arquivo `functions.php`:

```

1  <?php
2
3  /**
4   *      Atualizacao/Edicao de Cliente
5   */
6  function edit() {

```



```

18-11-2017 $id = $_GET['id'];
18-11-2017
18-11-2017 if (isset($_POST['customer'])) {
18-11-2017
18-11-2017     $customer = $_POST['customer'];
18-11-2017     $customer['modified'] = $now->format("Y-m-d H:i:s");
18-11-2017
18-11-2017     update('customers', $id, $customer);
18-11-2017     header('location: index.php');
18-11-2017 } else {
18-11-2017
18-11-2017     global $customer;
18-11-2017     $customer = find('customers', $id);
18-11-2017 }
18-11-2017 } else {
18-11-2017     header('location: index.php');
18-11-2017 }
18-11-2017 }

```

functions.php hosted with ❤ by GitHub

[view raw](#)

## Passo 5: Implemente o Formulário de Edição

Agora, crie um arquivo chamado edit.php na pasta customers. Esse arquivo vai ter a marcação do formulário de edição do cliente, que é quase igual à do cadastro:

```

18-11-2017 <?php
18-11-2017     require_once('functions.php');
18-11-2017     edit();
18-11-2017 ?>
18-11-2017
18-11-2017 <?php include(HEADER_TEMPLATE); ?>
18-11-2017
18-11-2017 <h2>Atualizar Cliente</h2>
18-11-2017
18-11-2017 <form action="edit.php?id=<?php echo $customer['id']; ?>" method="post">
18-11-2017     <hr />
18-11-2017     <div class="row">
18-11-2017         <div class="form-group col-md-7">
18-11-2017             <label for="name">Nome / Razão Social</label>
18-11-2017             <input type="text" class="form-control" name="customer['name']" value="<?php echo $customer['na
18-11-2017         </div>

```



```

</div>

<div class="form-group col-md-2">
    <label for="campo3">Data de Nascimento</label>
    <input type="text" class="form-control" name="customer['birthdate']" value="<?php echo $customer[
</div>
</div>
<div class="row">
    <div class="form-group col-md-5">
        <label for="campo1">Endereço</label>
        <input type="text" class="form-control" name="customer['address']" value="<?php echo $customer[
</div>

    <div class="form-group col-md-3">
        <label for="campo2">Bairro</label>
        <input type="text" class="form-control" name="customer['hood']" value="<?php echo $customer['ho
</div>

    <div class="form-group col-md-2">
        <label for="campo3">CEP</label>
        <input type="text" class="form-control" name="customer['zip_code']" value="<?php echo $customer
</div>

    <div class="form-group col-md-2">
        <label for="campo3">Data de Cadastro</label>
        <input type="text" class="form-control" name="customer['created']" disabled value="<?php echo $
</div>
</div>
<div class="row">
    <div class="form-group col-md-3">
        <label for="campo1">Município</label>
        <input type="text" class="form-control" name="customer['city']" value="<?php echo $customer['ci
</div>

    <div class="form-group col-md-2">
        <label for="campo2">Telefone</label>
        <input type="text" class="form-control" name="customer['phone']" value="<?php echo $customer['p
</div>

    <div class="form-group col-md-2">
        <label for="campo3">Celular</label>
        <input type="text" class="form-control" name="customer['mobile']" value="<?php echo $customer['
</div>

    <div class="form-group col-md-1">
        <label for="campo3">UF</label>

```



```

<label for="campo3">Inscrição Estadual</label>
<input type="text" class="form-control" name="customer['ie']" value="<?php echo $customer['ie']
</div>

<div class="form-group col-md-2">
<label for="campo3">UF</label>
<input type="text" class="form-control">
</div>
</div>
<div id="actions" class="row">
<div class="col-md-12">
<button type="submit" class="btn btn-primary">Salvar</button>
<a href="index.php" class="btn btn-default">Cancelar</a>
</div>
</div>
</form>

<?php include(FOOTER_TEMPLATE); ?>

```

edit.php hosted with ❤ by GitHub

[view raw](#)

## Passo 6: Crie a Função de Atualização no BD

E no arquivo database.php, implemente a função que faz a atualização de um registro no banco de dados:

```

<?php
/**
 * Atualiza um registro em uma tabela, por ID
 */
function update($table = null, $id = 0, $data = null) {

    $database = open_database();

    $items = null;

    foreach ($data as $key => $value) {
        $items .= trim($key, "'") . "='$value',";
    }

    // remove a ultima virgula
    $items = rtrim($items, ',');
}

```





```

try {
    $database->query($sql);

    $_SESSION['message'] = 'Registro atualizado com sucesso.';
    $_SESSION['type'] = 'success';

} catch (Exception $e) {

    $_SESSION['message'] = 'Nao foi possivel realizar a operacao.';
    $_SESSION['type'] = 'danger';
}

close_database($database);
}

```

database.php hosted with ❤ by GitHub

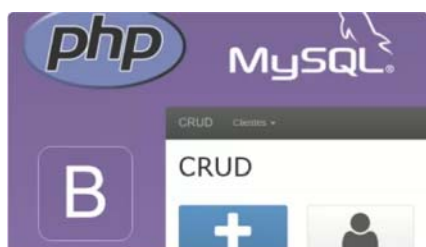
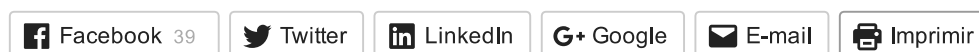
[view raw](#)

## Próximos Passos...

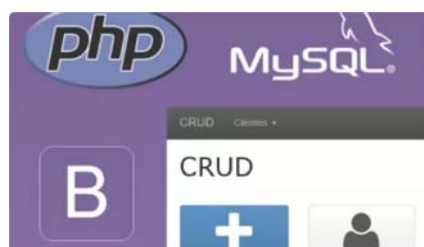
Há várias melhorias possíveis nesse código. Mas, como este é um tutorial para você começar a entender o CRUD, aos poucos será possível adicionar outras funcionalidades e melhorar a segurança e o reaproveitamento de funções.

No próximo tutorial, vamos ver a parte de exibição dos detalhes de um registro (Details).

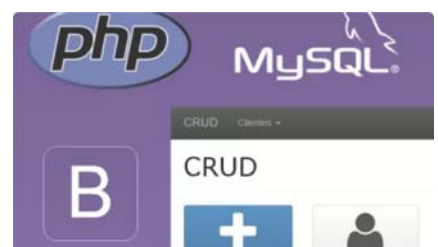
### Compartilhe:



Aprenda a Fazer um CRUD com Bootstrap, PHP & MySQL



CRUD com Bootstrap, PHP & MySQL – Parte V



CRUD com Bootstrap, PHP & MySQL – Parte III

← JavaScript: Trabalhando com Funções

CRUD com Bootstrap, PHP & MySQL –  
Parte V →

