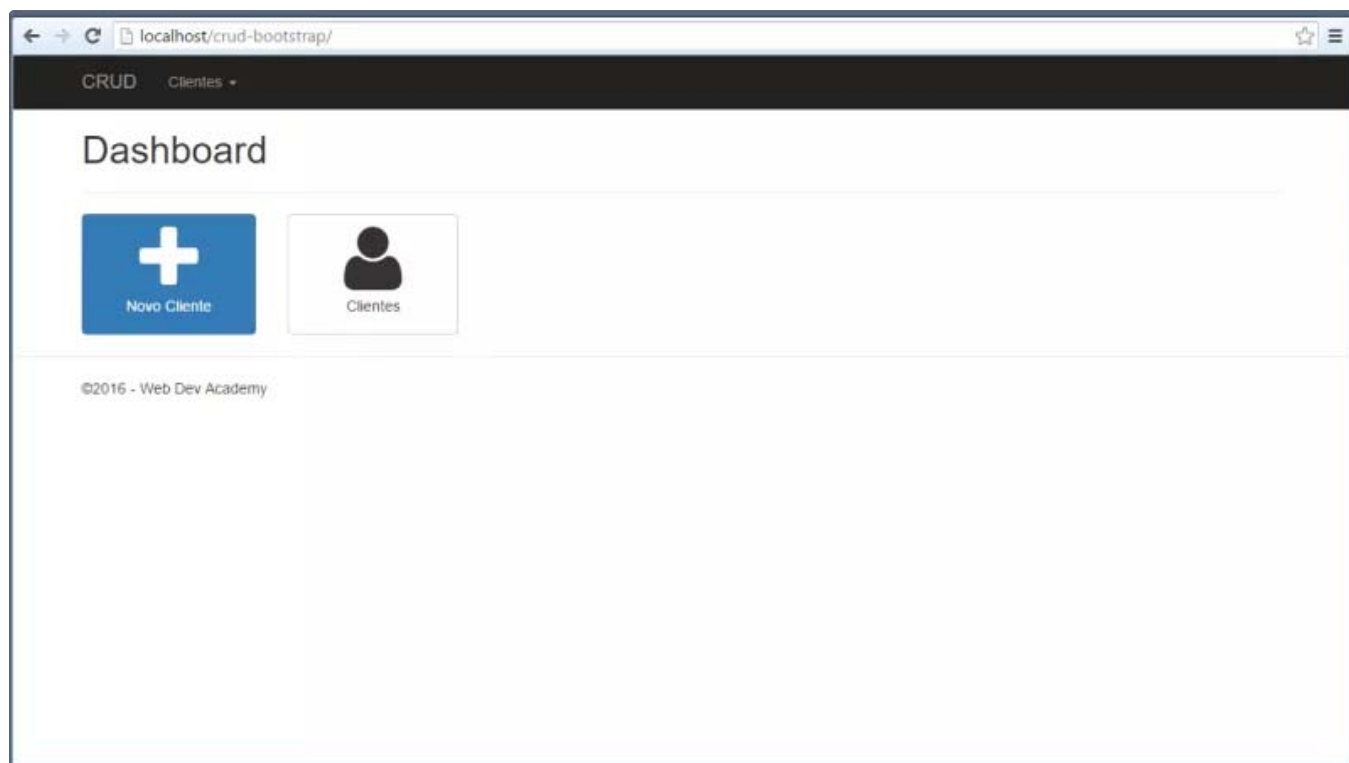


CRUD com Bootstrap, PHP & MySQL – Parte II

Sempre que você for fazer um sistema com PHP, ou até mesmo um site, você deve pensar em reaproveitamento de código. Ao fazer isso você aumenta sua produtividade, e evita gerar *bugs*. O ideal é que você use templates e funções sempre que possível.

Neste tutorial, você vai ver como criar os arquivos de template do cabeçalho (header) e do rodapé (footer) para usar em todo o projeto. Depois disso, vamos criar uma página inicial com os botões de acesso para as funcionalidades do sistema.



Ao final deste tutorial, você vai ter uma estrutura básica de templates para reaproveitar no projeto e, também, a página inicial que vai servir como *dashboard*, onde o usuário vai poder acessar as funcionalidades do sistema, e do CRUD.

criou o banco de dados do CRUD.

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.

Passo 1: Crie o Template do Header

Primeiro de tudo, crie um arquivo chamado **header.php** na pasta **/inc** do seu projeto. Depois coloque esta marcação:

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">
    <title>CRUD com Bootstrap</title>
    <meta name="description" content="">
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <link rel="stylesheet" href="<?php echo BASEURL; ?>css/bootstrap.min.css">
    <style>
        body {
            padding-top: 50px;
            padding-bottom: 20px;
        }
    </style>
    <link rel="stylesheet" href="<?php echo BASEURL; ?>css/style.css">
    <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/font-awesome/4.6.2/css/font-awesome.min.css">
</head>
<body>

    <nav class="navbar navbar-inverse navbar-fixed-top" role="navigation">
        <div class="container">
            <div class="navbar-header">
                <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#
                    <span class="sr-only">Toggle navigation</span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                    <span class="icon-bar"></span>
                </button>
                <a href="<?php echo BASEURL; ?>index.php" class="navbar-brand">CRUD</a>
            </div>
            <div id="navbar" class="navbar-collapse collapse">
                <ul class="nav navbar-nav">
                    <li class="dropdown">
```



```

39         <ul class="dropdown-menu">
40             <li><a href="<?php echo BASEURL; ?>customers">Gerenciar Clientes</a></li>
41             <li><a href="<?php echo BASEURL; ?>customers/add.php">Novo Cliente</a></li>
42         </ul>
43     </li>
44 </ul>
45 </div><!--/.navbar-collapse -->
46 </div>
47 </nav>
48
49 <main class="container">

```

header.php hosted with ❤ by GitHub

[view raw](#)

Bastante coisa, né?!

O que essa marcação faz é o definir início de uma página HTML básica, já usando o Bootstrap. E, também, vamos usar o Font Awesome, que é uma biblioteca de ícones, para os nossos botões.

Observe que em várias linhas tem o seguinte comando:

```
<?php echo BASEURL; ?>
```

Você deve se lembrar que a **constante BASEURL** foi definida no *config.php*, certo?!

Sempre que você for montar um link, você deve usar essa constante, já que ela guarda o endereço correto dessa nossa aplicação web no servidor.

Esse header também cria um menu (a partir da linha 22), que ficará no topo da página. Eu até adicionei dois links para os futuros “módulos” do CRUD:

```

<li><a href="<?php echo BASEURL; ?>customers">Gerenciar Clientes</a></li>
<li><a href="<?php echo BASEURL; ?>customers/add.php">Novo Cliente</a></li>

```

Essas páginas ainda não existem. Vamos criar nos próximos tutoriais. Mas já deixe os links aí, esperando pelo código do “módulo de clientes”.

Passo 2: Crie o Template do Footer



```

2      </main> <!-- /container -->
3
4      <hr>
5      <footer class="container">
6          <p>&copy;2016 - Web Dev Academy</p>
7      </footer>
8
9      <script src="//ajax.googleapis.com/ajax/libs/jquery/1.11.2/jquery.min.js"></script>
10     <script>window.jQuery || document.write('<script src="<?php echo BASEURL; ?>js/jquery-1.11.2.min.
11
12     <script src="<?php echo BASEURL; ?>js/bootstrap.min.js"></script>
13
14     <script src="<?php echo BASEURL; ?>js/main.js"></script>
15 </body>
16 </html>

```

footer.php hosted with ❤ by GitHub

[view raw](#)

Este aqui é mais simples.

O que essa marcação faz é “fechar” a página e criar um rodapé. Além disso, ela faz a referência aos arquivos JavaScript do Bootstrap e do jQuery.

Chamamos os `.js` ao final, para manter a performance da página.

Veja que eu também usei o `BASEURL` para fazer os links.

Passo 3: Altere o config.php

Abra o arquivo `config.php` e adicione as constantes para os templates no seu arquivo (as linhas 23 e 24, abaixo). Ele deve ficar assim:

```

1  <?php
2  /** O nome do banco de dados*/
3  define('DB_NAME', 'wda_crud');
4  /** Usuário do banco de dados MySQL */
5  define('DB_USER', 'root');
6  /** Senha do banco de dados MySQL */
7  define('DB_PASSWORD', '');
8  /** nome do host do MySQL */
9  define('DB_HOST', 'localhost');
10 /** caminho absoluto para a pasta do sistema */
11 if ( !defined('ABSPATH') )
12     define('ABSPATH', dirname(__FILE__) . '/');
13

```



```

17
18 /** caminho do arquivo de banco de dados */
19 if ( !defined('DBAPI') )
20     define('DBAPI', ABSPATH . 'inc/database.php');
21
22 /** caminhos dos templates de header e footer */
23 define('HEADER_TEMPLATE', ABSPATH . 'inc/header.php');
24 define('FOOTER_TEMPLATE', ABSPATH . 'inc/footer.php');
25

```

config.php hosted with ❤ by GitHub

[view raw](#)

A partir daqui, sempre que você criar uma página nova, você pode usar essas constantes `HEADER_TEMPLATE` e `FOOTER_TEMPLATE` para importar o topo e o final da página.

Isso evita muito a repetição de código, e economizará seu tempo e esforço.

Lembre-se que para usar esses templates, você precisará sempre requisitar o *config.php*, usando a função `require_once`.

Vamos ver isso no próximo passo...

Passo 4: Crie a Página Inicial

Agora, altere o arquivo *index.php* na **pasta principal** do seu projeto. E coloque esta marcação:

```

<?php require_once 'config.php'; ?>
<?php require_once DBAPI; ?>

<?php include(HEADER_TEMPLATE); ?>
<?php $db = open_database(); ?>

<h1>Dashboard</h1>
<hr />

<?php if ($db) : ?>

<div class="row">
    <div class="col-xs-6 col-sm-3 col-md-2">
        <a href="customers/add.php" class="btn btn-primary">
            <div class="row">
                <div class="col-xs-12 text-center">
                    <i class="fa fa-plus fa-5x"></i>
                </div>
            </div>
        </div>
    </div>
</div>

```



```

        </a>
    </div>

    <div class="col-xs-6 col-sm-3 col-md-2">
        <a href="customers" class="btn btn-default">
            <div class="row">
                <div class="col-xs-12 text-center">
                    <i class="fa fa-user fa-5x"></i>
                </div>
                <div class="col-xs-12 text-center">
                    <p>Clientes</p>
                </div>
            </div>
        </a>
    </div>
</div>

<?php else : ?>
    <div class="alert alert-danger" role="alert">
        <p><strong>ERRO:</strong> Não foi possível Conectar ao Banco de Dados!</p>
    </div>

<?php endif; ?>

<?php include(FOOTER_TEMPLATE); ?>

```

index.php hosted with ❤ by GitHub

[view raw](#)

As primeiras linhas fazem a inclusão do arquivo de configuração e da camada de banco de dados.

Na linha 4, temos o seguinte comando:

```
<?php include(HEADER_TEMPLATE); ?>
```

É ele que faz a importação do nosso template de header para a página, e traz toda aquela marcação em HTML. Assim, você não precisa escrever o topo da página em cada script.

Depois disso, é que começa a página em si. Coloquei um título simples, e um grid que vai manter os botões do *dashboard*. Esses botões usam o componente do Bootstrap e o ícone vem do Font Awesome.

Por último, usei o comando para importar o template footer da página:



<http://localhost/crud-bootstrap>

Até aqui, seu projeto deve estar assim:

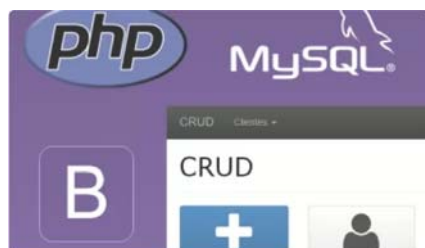
- crud-bootstrap-php
 - css
 - fonts
 - inc
 - database.php
 - footer.php
 - header.php
 - js
 - config.php
 - index.php

Próximos Passos...

No próximo tutorial, vamos implementar a tela de listagem, que permite acessar as principais funções do CRUD, e que exibe todos os registros cadastrados.

Até a próxima!

Compartilhe:



Aprenda a Fazer um CRUD com Bootstrap, PHP & MySQL



CRUD com Bootstrap, PHP & MySQL – Parte VI



CRUD com Bootstrap, PHP & MySQL - Parte I

← CRUD com Bootstrap, PHP & MySQL – Parte I

CRUD com Bootstrap, PHP & MySQL – Parte III →

