

Arquiteturas web e CodeIgniter

Desenvolvimento web para bioinformática

Aula 7/8

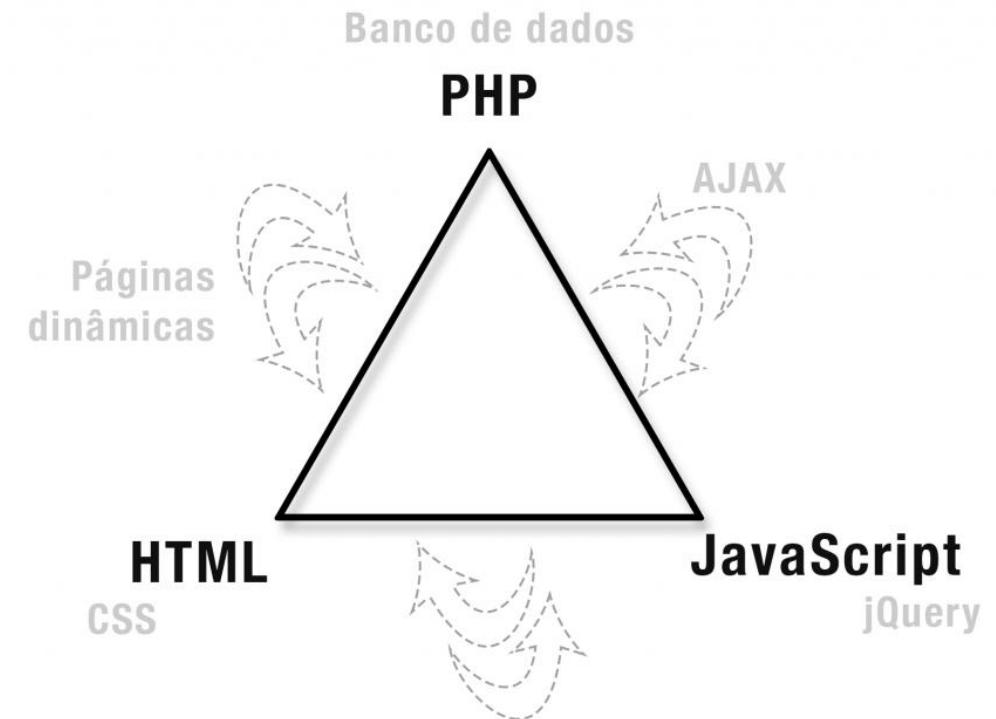
Avaliação

- **Entrega do trabalho final**
 - **Prazo:** 7 dias
 - Código ou vídeo (5 a 10 min) – deve enviar apenas o link do youtube
 - Enviar por e-mail (para o Rafael)
- **Apresentação na sexta (opcional)**
 - Quem apresentar na sexta não precisa enviar o código/vídeo por e-mail

O dogma central do desenvolvimento web

na antiguidade

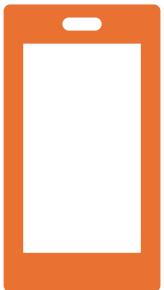
Back-end



Front-end

Entretanto, isso está desatualizado

Arquiteturas de software



Monolitos

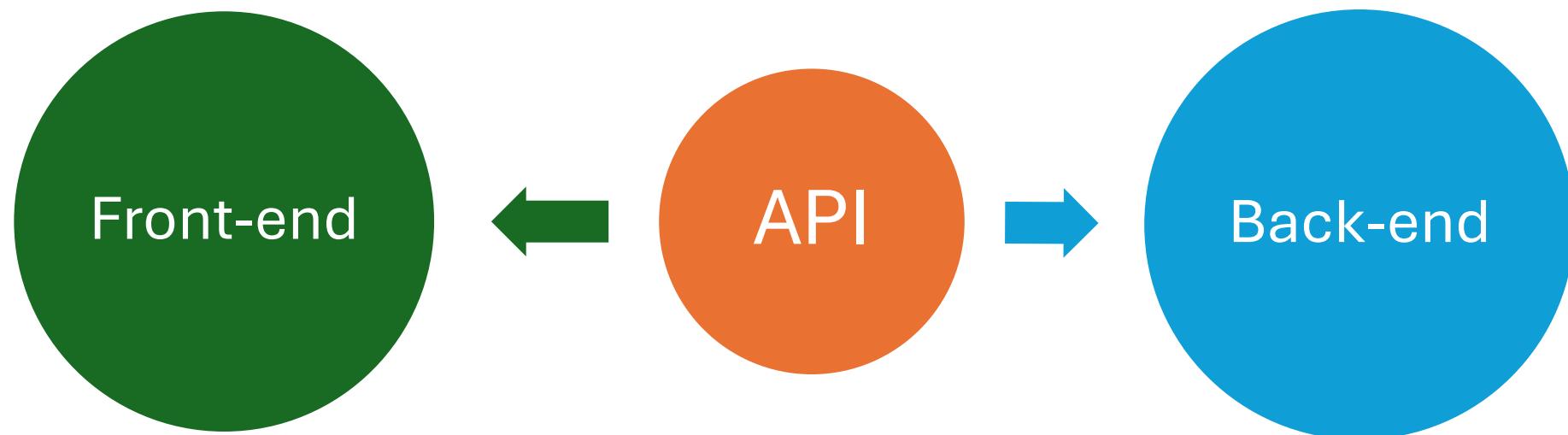
aplicativo único e indivisível



Microsserviços

conjunto de serviços independentes e
pequenos que se comunicam entre si

Arquiteturas modernas



Por que precisamos aprender?

- Suporte a sistemas legado
- Desenvolvimento mais rápido

PHP

- Linguagem de programação usada para criação de páginas dinâmicas
 - <https://www.php.net/>
- Requer um servidor web (Apache) com interpretador PHP
 - https://www.apachefriends.org/pt_br/
- Linguagem usada em mais de 70% dos sites da internet
 - Uns dizem que está pra morrer faz uns 10 anos
 - Outros dizem que é imortal
- Aprendizado rápido
 - Excelente para fazer gambiarra



PHP

- **Linguagem server-side:** PHP é executado no servidor e gera HTML enviado ao navegador.
- **Extensão:** “.php”
- **Sintaxe simples:** <?php ... ?> para delimitar código PHP dentro de arquivos HTML.
- **Variáveis:** iniciam com \$

```
<div class="container">
    <div class="row">
        <div class="col">
            <?php echo "Sou um código em PHP"; ?>
        </div>
    </div>
</div>
```

Sintaxe reduzida: <?=“Olá”?>

```
<?php
    $nome = "José";
```

PHP

- **Arrays:** coleções de dados indexados numericamente ou associativamente.

```
<?php  
| $produtos = ["Pão", "Leite", "Ovos"];
```

- **Funções:** blocos reutilizáveis de código (function nomeFuncao() { ... }).

```
<?php  
| // declaração  
| function ola(){  
| | echo "Olá!";  
| }  
  
| // execução  
| ola()
```

PHP

- **Controle de fluxo:** if, else, switch, while, for, etc.

```
<?php
    if($x > 0){
        echo "$x maior que zero";
    }
    else{
        echo "$x menor que zero";
    }
```

PHP – formulários

- **Formulários:** PHP lida com dados enviados por GET ou POST.

index.html

```
<form action="processa.php" method="POST">
    <input type="text" name="nome" placeholder="Digite seu nome">

    <input type="submit" value="Enviar">
</form>
```

processa.php

```
1  <?php
2
3  // recebe os dados enviados pelo formulário
4  $nome = $_POST['nome'];
5
6  echo "O nome enviado pelo formulário é $nome";
7
```

PHP – outras características

- **Integração com bancos de dados:** comum usar MySQL via mysqli ou PDO.
- **Programação orientada a objetos (POO):** suporte a classes, objetos, herança e encapsulamento.
- **Superglobais:** variáveis especiais como \$_POST, \$_GET, \$_SESSION, \$_FILES.
- **Sessions e Cookies:** usados para armazenar dados entre requisições.

Exemplos PHP

- Mostrar exemplos de página dinâmica com PHP
 - **Exemplo 1:** site dinâmico de receitas de bolo
 - **Exemplo 2:** executando Python via PHP

Codelgniter



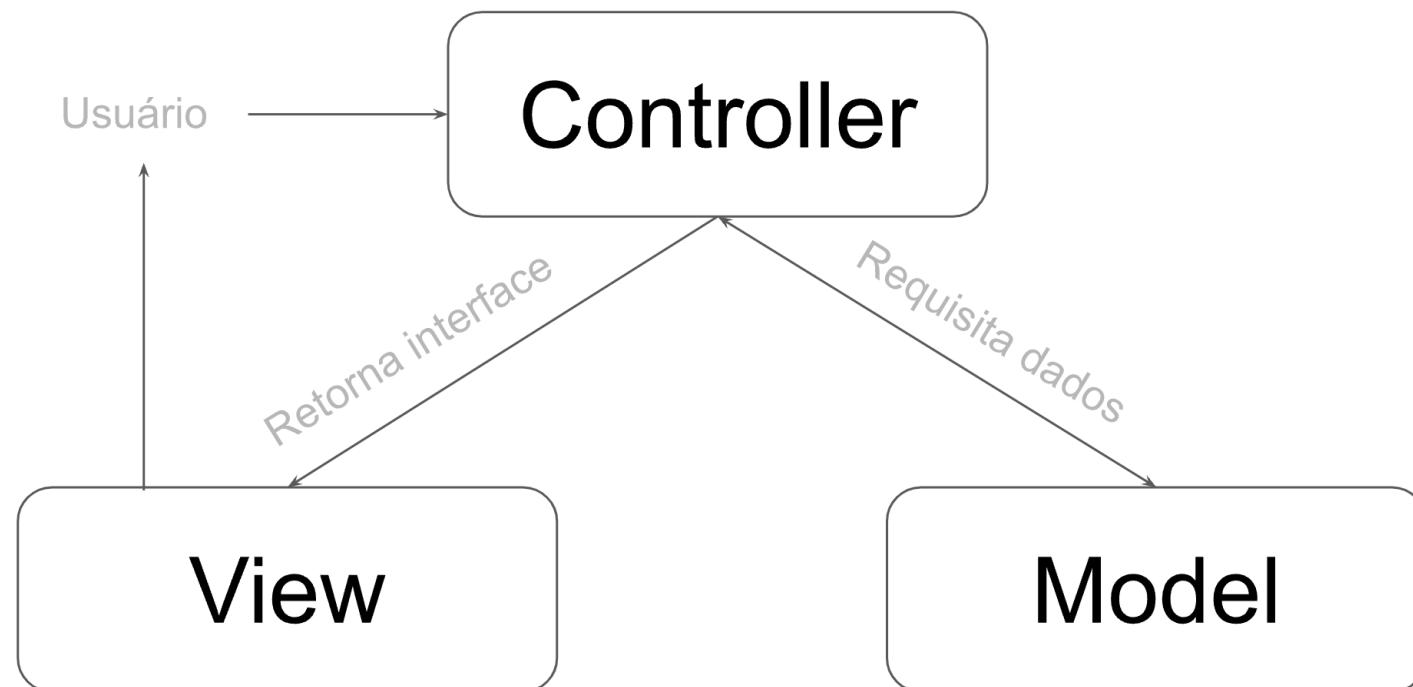
CodeIgniter

- Framework PHP simples e leve
 - Código fonte 1,1 MB
 - <https://www.codeigniter.com/>

Aplicações em bioinformática

Do planejamento à produção do MVP em menos de 24 h

Arquitetura MVC



Instalando o CodeIgniter

Existem duas formas principais de instalar o CodeIgniter:

1. **Instalação manual:** Baixe os arquivos do repositório oficial:
<https://github.com/CodeIgniter4/framework/releases/latest>
2. **Instalação via Composer:** Utilize o comando:

```
composer create-project codeigniter4/appstarter meu-site
```

XAMPP

XAMPP 8.2.4-0

Welcome Manage Servers Server Events

Welcome to XAMPP 8.2.4
Rev. 0

Follow XAMPP

Go to Application
Open Application Folder
Visit Apache Friends
Get Support

The image shows the XAMPP 8.2.4-0 application window. On the left, the 'Welcome' tab is selected, displaying the XAMPP logo and social media links for Facebook and Twitter. Below the logo are four buttons: 'Go to Application', 'Open Application Folder' (which is highlighted), 'Visit Apache Friends', and 'Get Support'. On the right, a file browser window is open, showing the contents of the 'htdocs' folder within the 'xamppfiles' directory. The 'htdocs' folder is currently selected. The browser interface includes standard file operations like copy, paste, and search.

File/Folder	Description
apache2	Apache server configuration files
bin	Binary executables
build	Build-related files
cgi-bin	CGI scripts
ctlscript.sh	Control script for the server
docs	Documentation files
error	Error logs and files
etc	Configuration files for the server
htdocs	The main web root directory, currently selected
icons	Icon files
img	Image files
include	Include files
info	Information files
lib	Library files
libexec	Libexec files
licenses	Licenses and legal files
logs	Log files

xamppfiles > htdocs

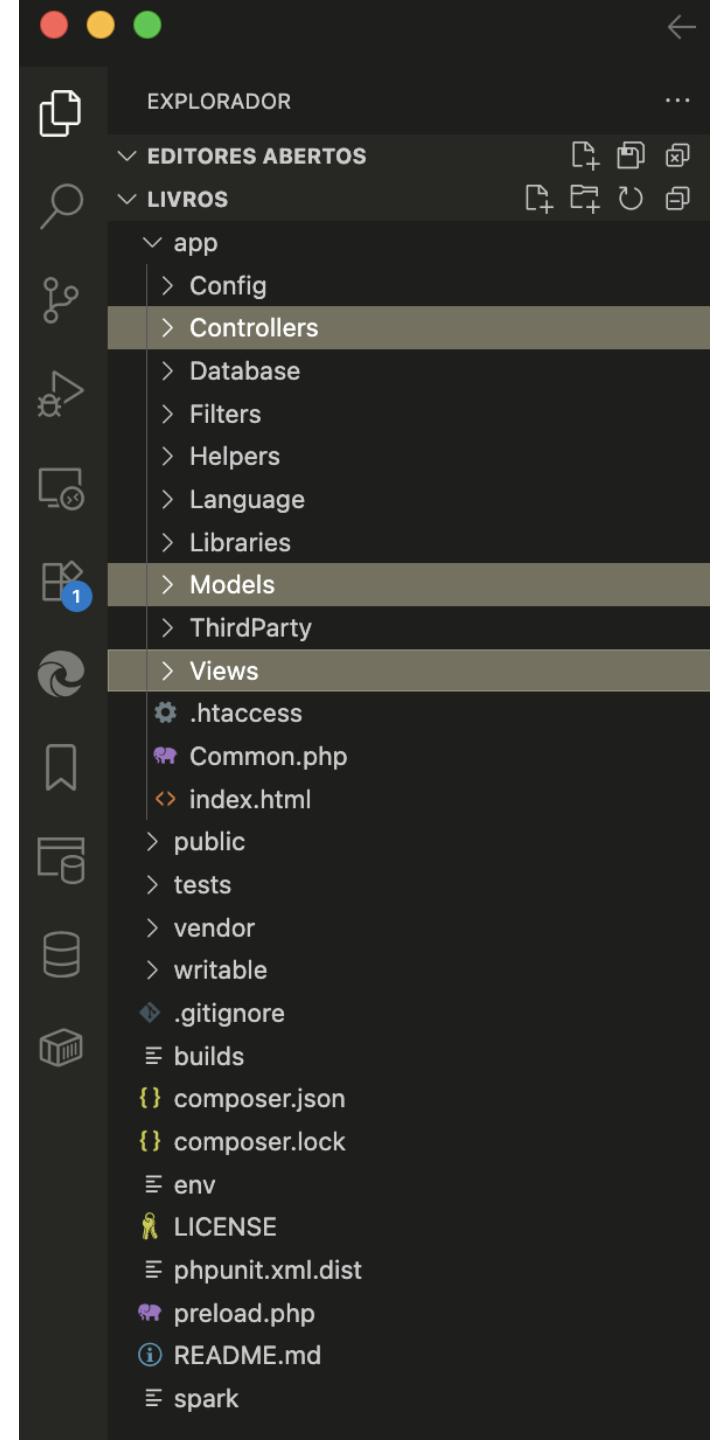
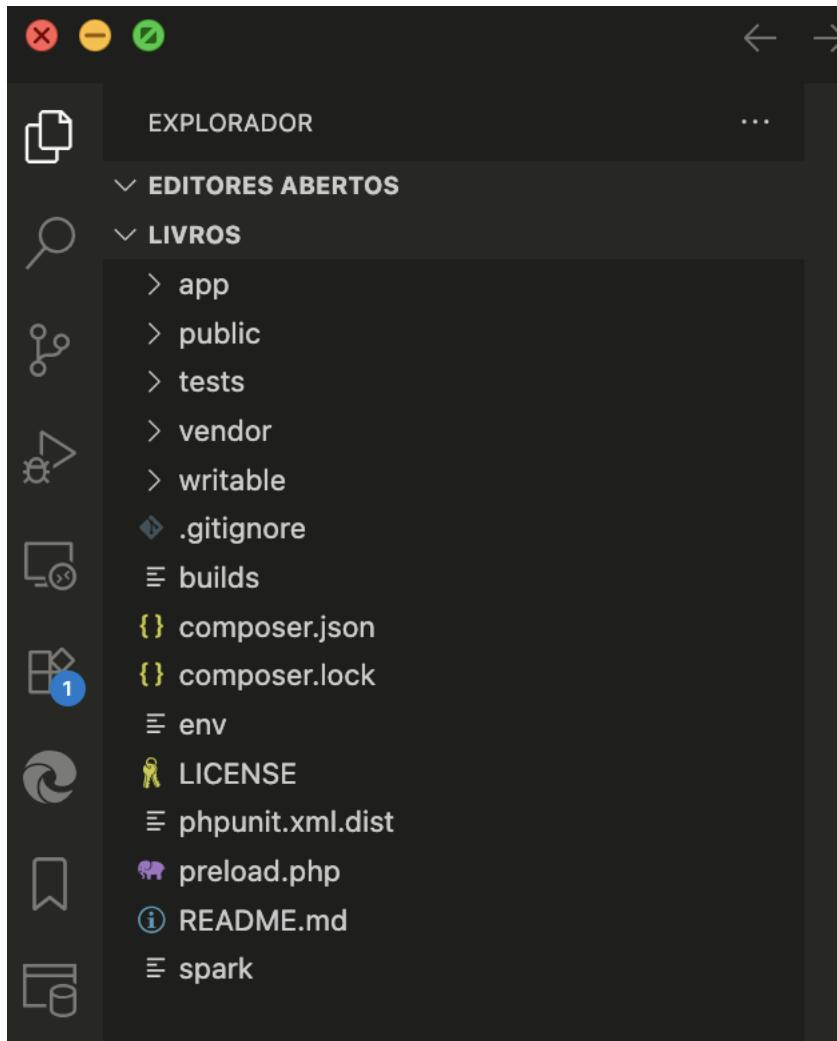
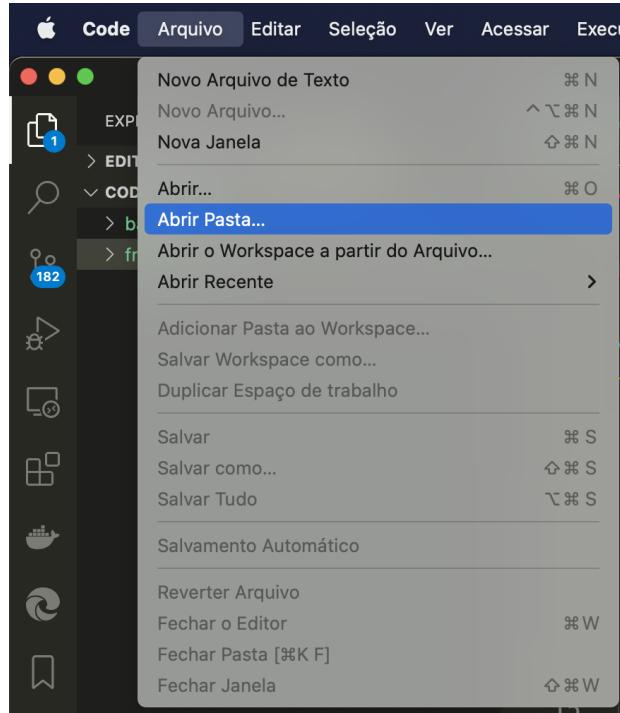
```
(base) diego@hipogrifo-341 GitHub % composer create-project codeigniter4/appstarter livros|
```

```
(base) diego@hipogrifo-341 GitHub % cd livros
(base) diego@hipogrifo-341 livros % php spark serve
```

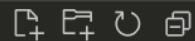
```
CodeIgniter v4.6.1 Command Line Tool - Server Time: 2025-07-10 02:32:25 UTC+00:00
CodeIgniter development server started on http://localhost:8080
Press Control-C to stop.
[Wed Jul  9 23:32:25 2025] PHP 8.4.4 Development Server (http://localhost:8080) started
```

```
Welcome to CodeIgniter 4!
localhost:8080
```

The image shows a Mac OS X desktop environment. At the top, there is a system status bar with three colored dots (red, yellow, green) on the left and a folder icon labeled "GitHub — -zsh" followed by the path "~Documents/GitHub — -zsh" on the right. Below the status bar is a terminal window titled "GitHub — -zsh" containing command-line text. To the right of the terminal is a browser window titled "livros — php · php spark serve" showing the path "~Documents/GitHub/livros — php · php spark serve". The browser window displays the "Welcome to CodeIgniter 4!" page. At the bottom of the screen, the Mac OS X dock is visible, featuring icons for Finder, Mail, Safari, and other applications.



✓ LIVROS



✓ app

> Config
> Controllers
> Database
> Filters
> Helpers
> Language
> Libraries
> Models
> ThirdParty

✓ Views

> errors

✓ welcome_message.php

⚙ .htaccess

🐘 Common.php

▷ index.html

> public

> tests

> vendor

> writable

❖ .gitignore

≡ builds

{ } composer.json

{ } composer.lock

≡ env

🐘 LICENSE

≡ phpunit.xml.dist

🐘 preload.php

ⓘ README.md

≡ spark

🐘 welcome_message.php ×

app > Views >🐘 welcome_message.php > ...

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Meu primeiro site com CodeIgniter</title>
5      </head>
6      <body>
7          <p>Olá mundo, CodeIgniter!</p>
8          <p><?=date('Y')?></p>
9      </body>
10 </html>
```



Meu primeiro site com CodeIgniter

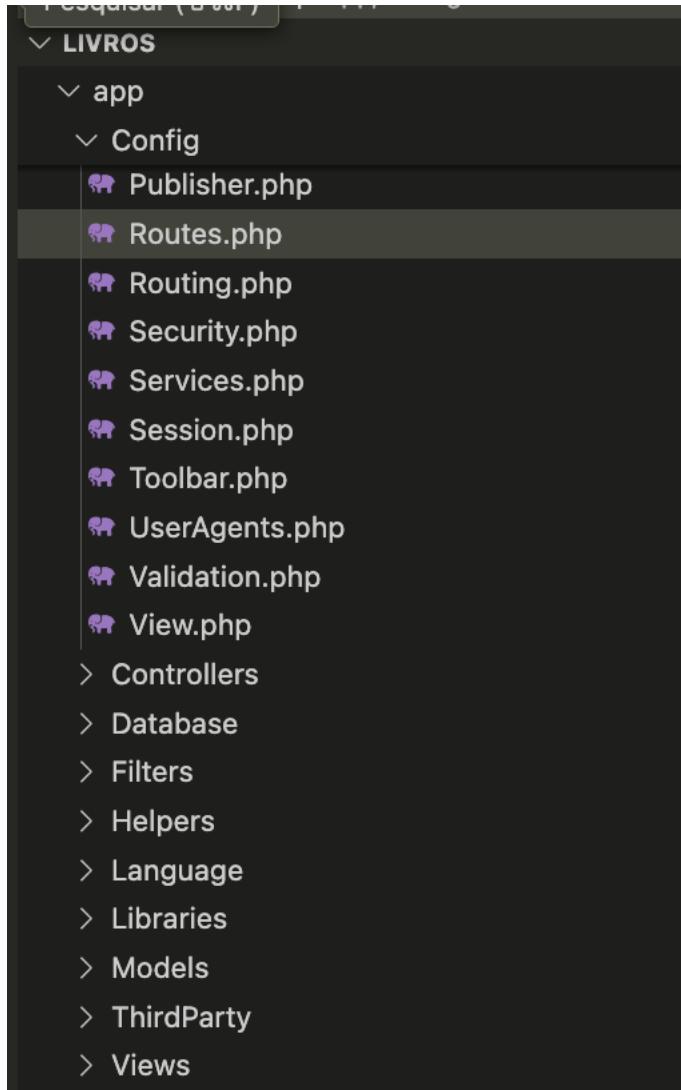


← → ⌂ ⓘ localhost:8080

Olá mundo, CodeIgniter!

2025

Rotas



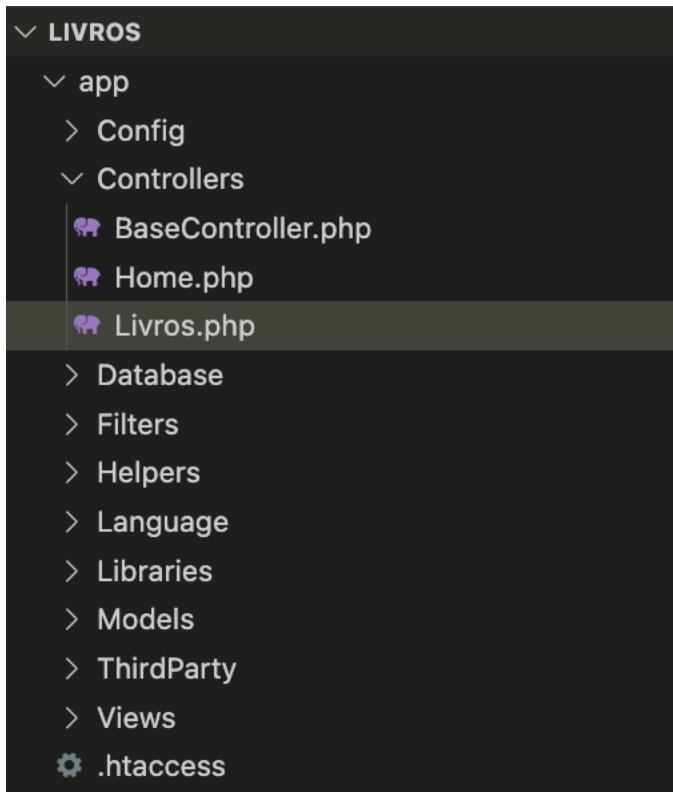
```
Routes.php × Livros.php livros.php
app > Config > Routes.php

1  <?php
2
3  use CodeIgniter\Router\RouteCollection;
4
5  /**
6   * @var RouteCollection $routes
7  */
8  $routes->get('/', 'Home::index');
9
10 $routes->get('/livros', 'Livros::listar');
```

- Rota /livros aponta para a função listar() dentro do controller **Livros**

Controller

- Crie o arquivo Livros.php

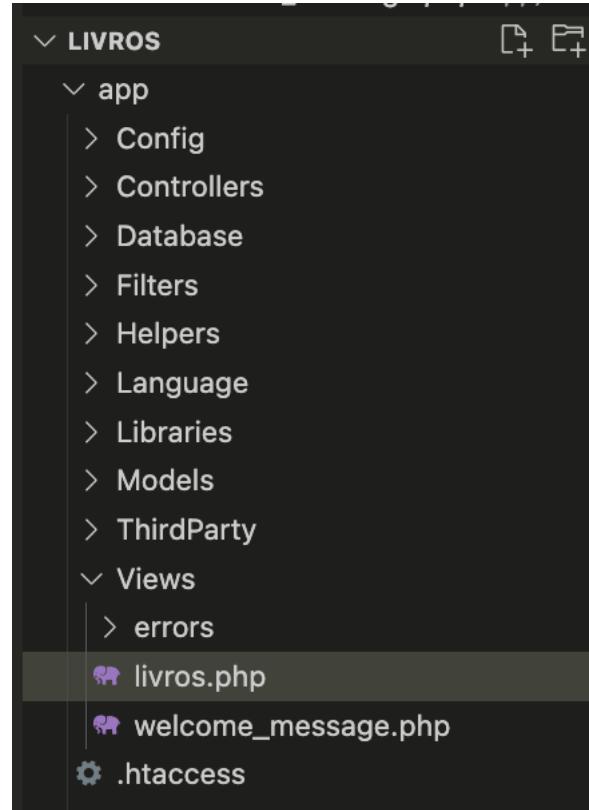


```
Routes.php Livros.php × Home.php
app > Controllers > Livros.php > ...
1 <?php
2
3 namespace App\Controllers;
4
5 class Livros extends BaseController
6 {
7     public function listar(): string
8     {
9         $data = []; // crie um array vazio
10        $data['livros'] = ["Harry Potter", "Senhos dos Anéis", "Crônicas de Nárnia"];
11        return view('livros', $data); // chame a view que irá exibir isto
12    }
13 }
14
```

The image shows a code editor window with a dark theme. The title bar says 'Routes.php', 'Livros.php ×', and 'Home.php'. The status bar indicates the path: 'app > Controllers > Livros.php > ...'. The code itself is a PHP class definition:

```
<?php
namespace App\Controllers;
class Livros extends BaseController
{
    public function listar(): string
    {
        $data = []; // crie um array vazio
        $data['livros'] = ["Harry Potter", "Senhos dos Anéis", "Crônicas de Nárnia"];
        return view('livros', $data); // chame a view que irá exibir isto
    }
}
```

View

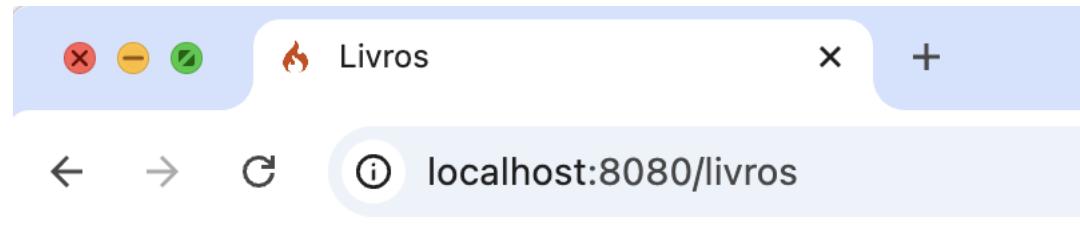


The screenshot shows a code editor with the following code:

```
Routes.php Livros.php livros.php X welcome_message.php

app > Views > livros.php > html > body > ul
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Livros</title>
5      </head>
6      <body>
7          <h1>Lista de livros</h1>
8          <ul>
9              <?php
10             foreach($livros as $livro){|
11                 echo "<li>$livro</li>";|
12             } ?>
13         </ul>
14     </body>
15 </html>
```

Acessando a rota

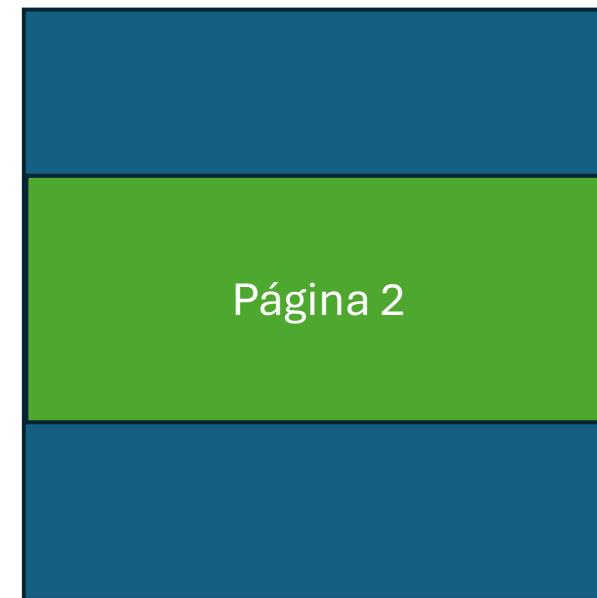
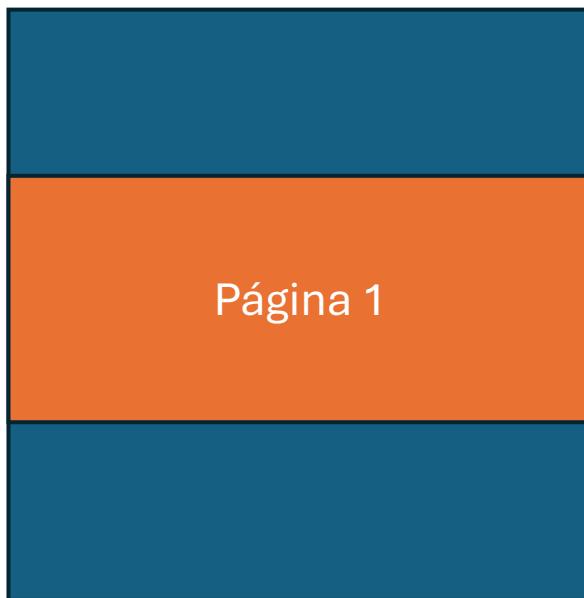


Lista de livros

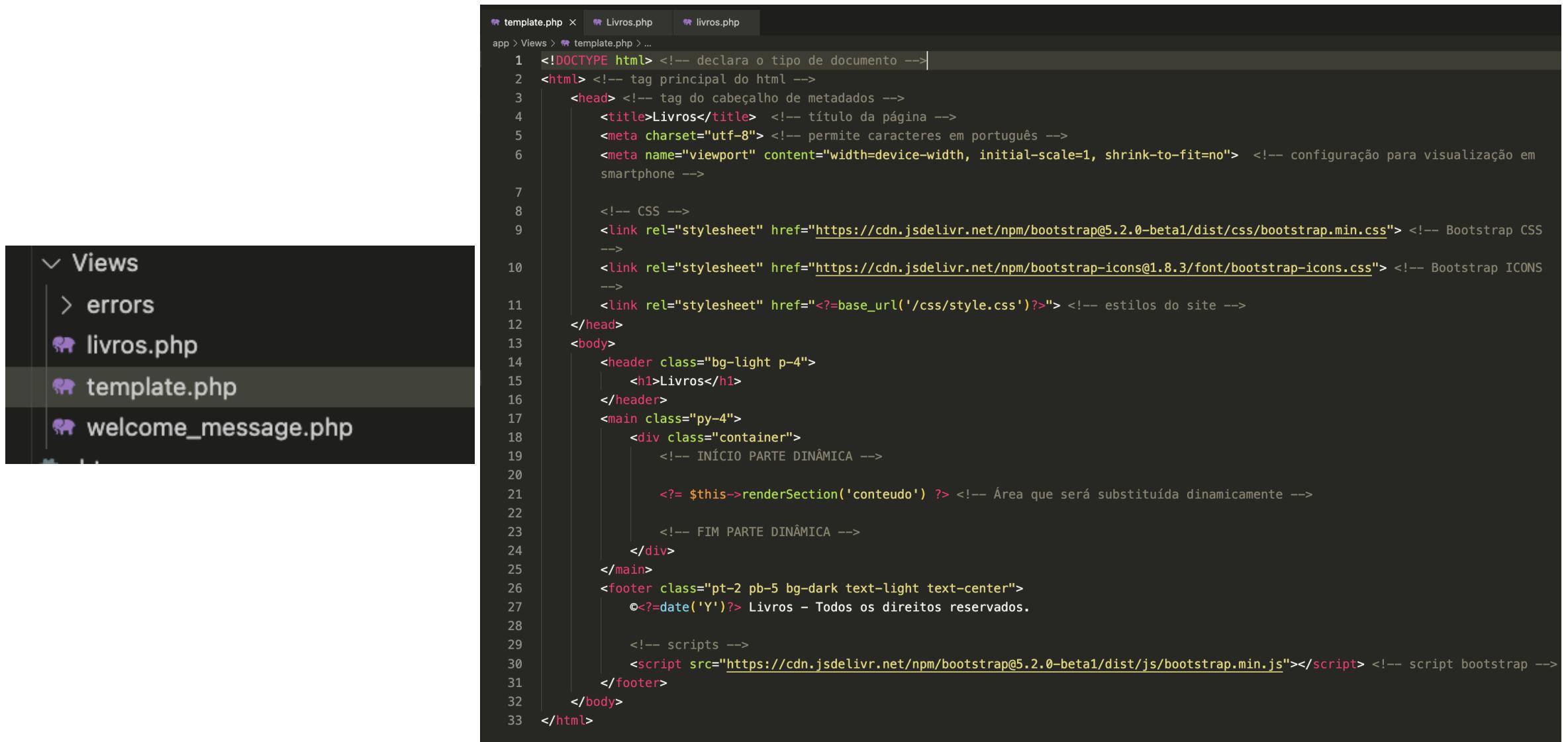
- Harry Potter
- Senhos dos Anéis
- Crônicas de Nárnia

Criando um template

- Um template é uma interface padrão que será reaproveitada em todas as páginas



Crie o arquivo app/Views/template.php



The image shows a code editor window and a file browser window side-by-side.

File Browser (Left):

- Views
 - > errors
 - livros.php
 - template.php
 - welcome_message.php

Code Editor (Right):

```
template.php Livros.php livros.php
app > Views > template.php > ...
1  <!DOCTYPE html> <!-- declara o tipo de documento -->
2  <html> <!-- tag principal do html -->
3  |   <head> <!-- tag do cabeçalho de metadados -->
4  |       <title>Livros</title> <!-- título da página -->
5  |       <meta charset="utf-8"> <!-- permite caracteres em português -->
6  |       <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no"> <!-- configuração para visualização em smartphone -->
7
8
9
10 <!-- CSS -->
11 <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-beta1/dist/css/bootstrap.min.css"> <!-- Bootstrap CSS -->
12 <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.8.3/font/bootstrap-icons.css"> <!-- Bootstrap ICONS -->
13 <link rel="stylesheet" href="=base_url('/css/style.css')?&gt;"&gt; &lt;!-- estilos do site --&gt;
14 &lt;/head&gt;
15 &lt;body&gt;
16     &lt;header class="bg-light p-4"&gt;
17         &lt;h1&gt;Livros&lt;/h1&gt;
18     &lt;/header&gt;
19     &lt;main class="py-4"&gt;
20         &lt;div class="container"&gt;
21             &lt;!-- INÍCIO PARTE DINÂMICA --&gt;
22
23             &lt;?= $this-&gt;renderSection('conteudo') ?&gt; &lt;!-- Área que será substituída dinamicamente --&gt;
24
25             &lt;!-- FIM PARTE DINÂMICA --&gt;
26         &lt;/div&gt;
27     &lt;/main&gt;
28     &lt;footer class="pt-2 pb-5 bg-dark text-light text-center"&gt;
29         ©&lt;?=date('Y')?&gt; Livros - Todos os direitos reservados.
30
31         &lt;!-- scripts --&gt;
32         &lt;script src="https://cdn.jsdelivr.net/npm/bootstrap@5.2.0-beta1/dist/js/bootstrap.min.js"&gt;&lt;/script&gt; &lt;!-- script bootstrap --&gt;
33     &lt;/footer&gt;
34 &lt;/body&gt;
35 &lt;/html&gt;</pre
```

Parte mais importante (template.php)

```
<main class="py-4">
    <div class="container">
        <!-- INÍCIO PARTE DINÂMICA -->

        <?= $this->renderSection('conteudo') ?> <!-- Área que será substituída dinamicamente -->

        <!-- FIM PARTE DINÂMICA -->
    </div>
</main>
```

Como usar um template nas suas views?

- Antes

The screenshot shows a code editor with a dark theme. The file path is 'app > Views > livros.php'. The code is as follows:

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Livros</title>
5      </head>
6      <body>
7          <h1>Lista de livros</h1>
8          <ul>
9              <?php
10             foreach($livros as $livro){
11                 echo "<li>$livro</li>";
12             } ?>
13         </ul>
14     </body>
15 </html>
```

- Depois

The screenshot shows a code editor with a dark theme. The file path is 'app > Views > livros.php'. The code is as follows:

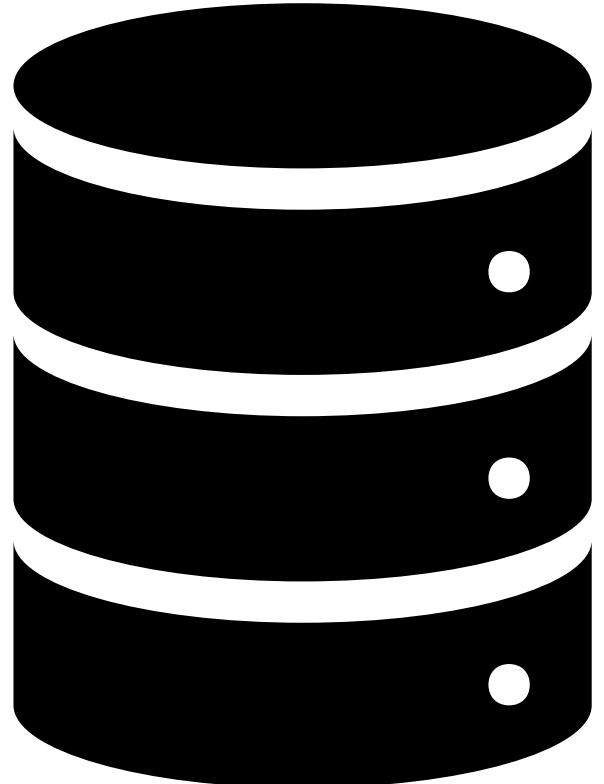
```
1  <?= $this->extend('template') ?>
2  <?= $this->section('conteudo') ?>
3
4  <h1>Lista de livros</h1>
5  <ul>
6      <?php
7      foreach($livros as $livro){
8          echo "<li>$livro</li>";
9      } ?>
10     </ul>
11
12 <?= $this->endSection() ?>
```

Como ficou o site?



Lista de livros

- Harry Potter
- Senhos dos Anéis
- Crônicas de Nárnia



Trabalhando com bancos de dados

- **Armazenamento estruturado de dados**
- Bancos de dados são sistemas que armazenam informações de forma organizada, facilitando o acesso, gerenciamento e atualização de dados.

Trabalhando com banco de dados

- **Sistemas Gerenciadores de Banco de Dados (SGBD)**
São softwares como MySQL, PostgreSQL ou SQLite que permitem criar, consultar e manipular bancos de dados com segurança e eficiência.

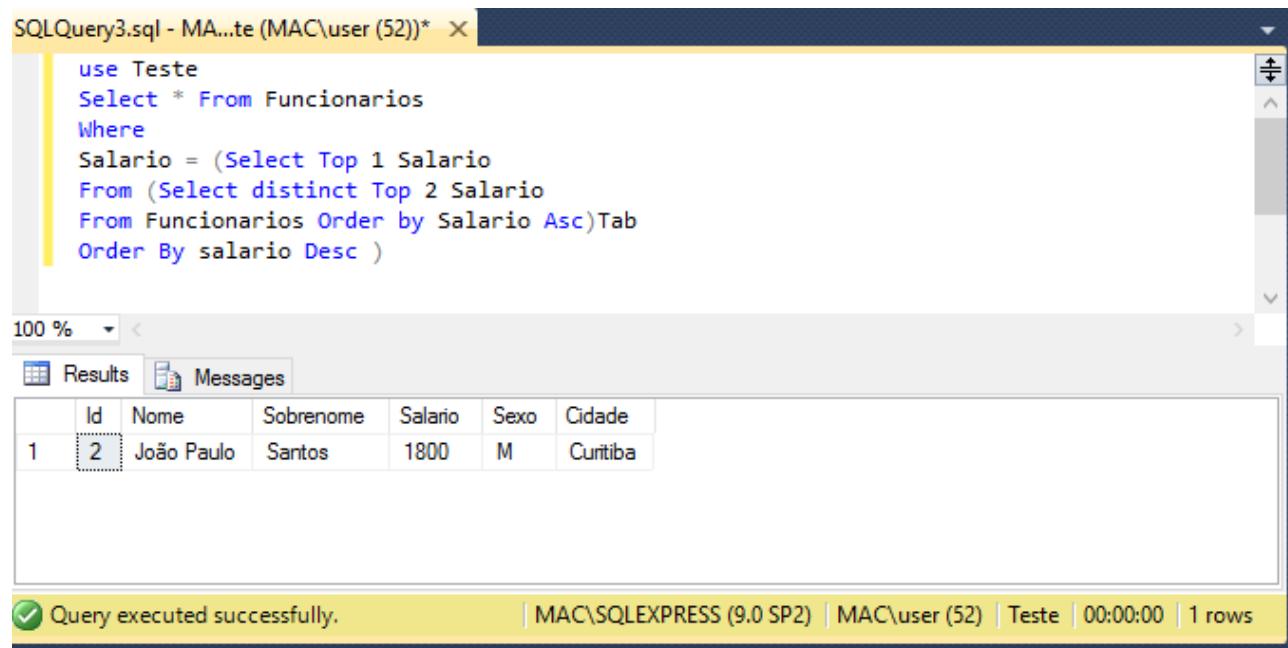


Trabalhando com banco de dados

- **Linguagem de consulta (SQL)**

A maioria dos bancos de dados relacionais utiliza SQL (*Structured Query Language*) para realizar operações como inserir, buscar, atualizar e excluir dados.

Exemplo de código SQL



The screenshot shows a SQL Server Management Studio (SSMS) interface. The top window is titled "SQLQuery3.sql - MA...te (MAC\user (52))". It contains the following SQL code:

```
use Teste
Select * From Funcionarios
Where
    Salario = (Select Top 1 Salario
    From (Select distinct Top 2 Salario
    From Funcionarios Order by Salario Asc)Tab
    Order By salario Desc )
```

Below the code window is a results grid showing one row of data:

	Id	Nome	Sobrenome	Salario	Sexo	Cidade
1	2	João	Paulo Santos	1800	M	Curitiba

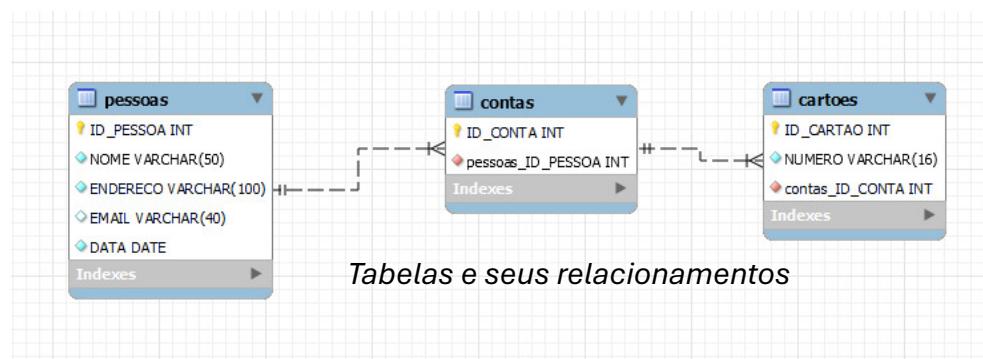
At the bottom of the screen, a status bar displays the message "Query executed successfully." and the connection information "MAC\SQLEXPRESS (9.0 SP2) | MAC\user (52) | Teste | 00:00:00 | 1 rows".

Trabalhando com banco de dados

- **Tipos de bancos de dados**

Existem diferentes tipos, como relacionais (tabelas), NoSQL (documentos, grafos, chave-valor), cada um ideal para diferentes aplicações e volumes de dados.

Bancos relacionais (SQL)



Tabelas e seus relacionamentos



Vs.

Bancos não relacionais (NoSQL)

json

```
{  
  "_id": "64fabc1234567890abcdef12",  
  "titulo": "Bioinformática para Iniciantes",  
  "autor": "Diego Mariano",  
  "ano": 2023,  
  "paginas": 250,  
  "categorias": ["Ciência", "Tecnologia", "Educação"]  
}
```



COMPARANDO O DESEMPENHO DE BANCOS DE DADOS NOSQL E RELACIONAIS MANIPULANDO DADOS BIOLÓGICOS

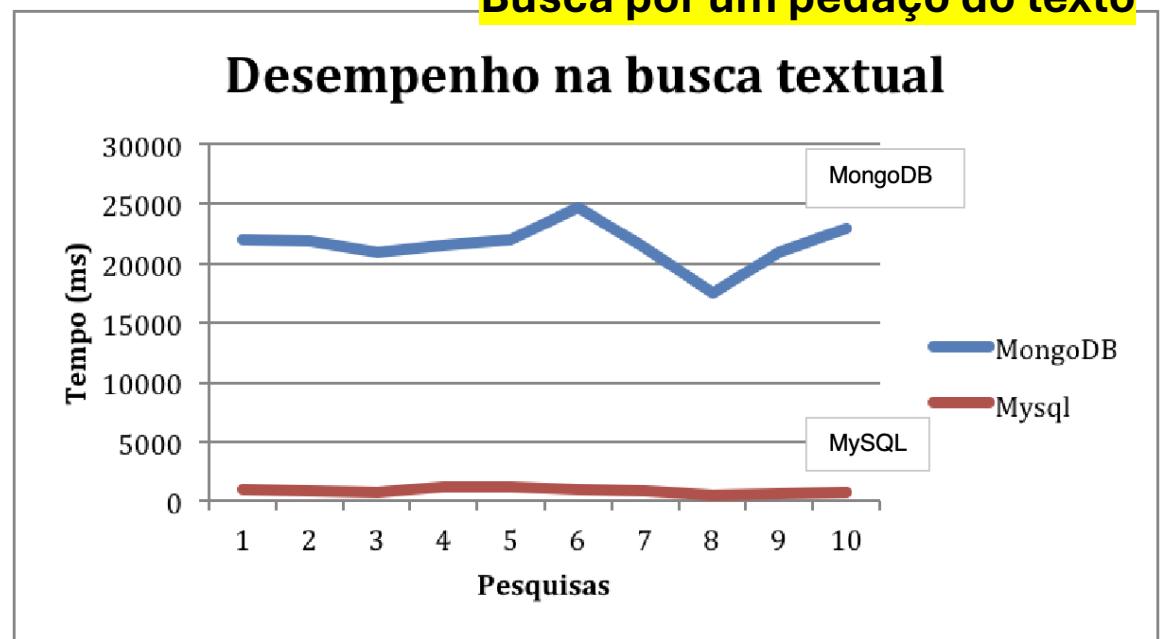
January 2012

Diego Mariano · Edgar Lacerda De Aguiar · Sandro Renato Dias

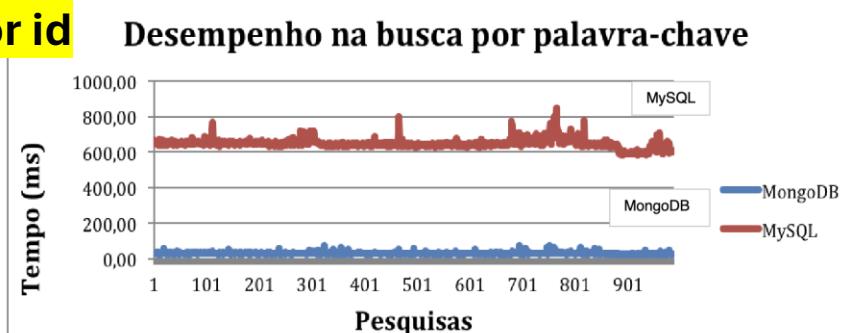
Inserção de dados



Busca por um pedaço do texto



Seleção por id

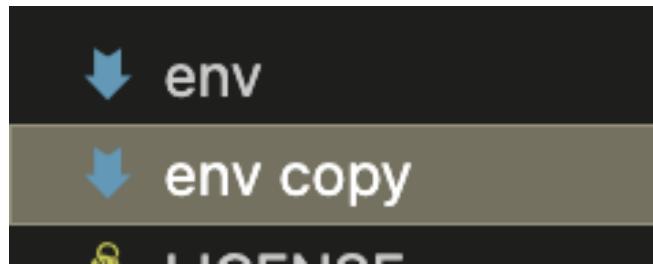


Vamos usar SQLite

- Um dos SGBDs mais utilizados no mundo
- Usado em aplicativos de celular
- Armazenamento baseado em arquivos
- Permite consultas SQL

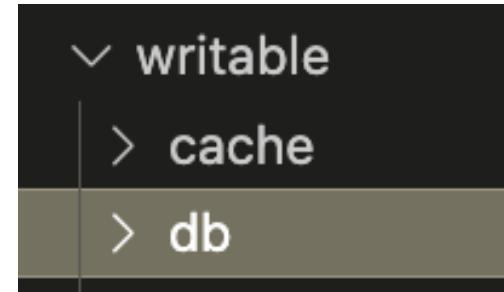


Primeiro, faça uma cópia do arquivo **env**

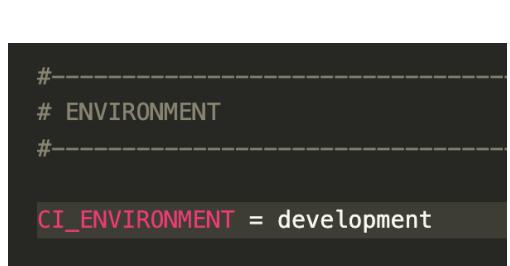


Renomeie para .env

Crie a pasta db em writable



Modifique o arquivo .env



```
29 #-----  
30 # DATABASE  
31 #-----  
32  
33 database.default.hostname = localhost  
34 database.default.database = ../writable/db/database.sqlite  
35 database.default.username =  
36 database.default.password =  
37 database.default.DBDriver = SQLite3  
38 # database.default.DBPrefix =  
39 # database.default.port = 3306  
40
```

Bancos de dados suportados oficialmente:

1. MySQL / MariaDB

- Driver: `MySQLi` (padrão)
- Uso comum em aplicações web



2. PostgreSQL

- Suporte robusto a dados complexos e JSON
- Driver: `Postgre`

3. SQLite3

- Leve e sem necessidade de servidor
- Ideal para testes e apps simples

4. SQL Server

- Suporte limitado ao Microsoft SQL Server
- Driver: `SQLSRV` (requer extensões do PHP)

Outras opções (com extensões ou integrações manuais):

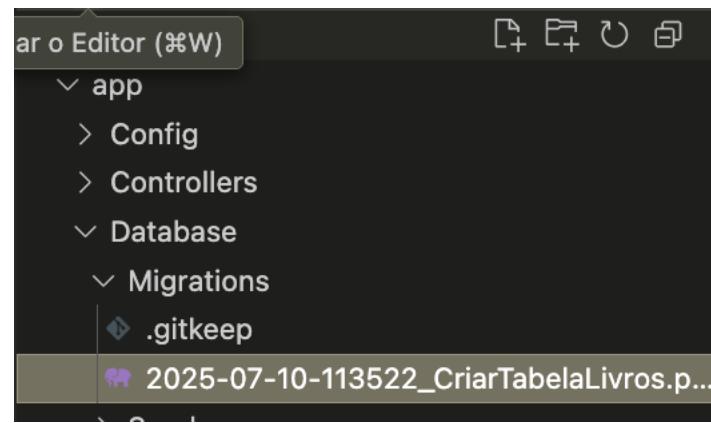
- **Oracle** (não suportado oficialmente, mas possível com bibliotecas externas)
- **MongoDB** (não relacional, exige uso de bibliotecas de terceiros como [jenssegers/mongodb](#), adaptadas para CodeIgniter)
- **Firebird**, **IBM DB2** e outros – com bastante esforço e bibliotecas externas.

Criando uma migration

- Uma **migration** (ou **migração**) é um recurso usado para **controlar a estrutura do banco de dados via código**

```
(base) diego@hipogrifo-341 livros % php spark make:migration CriarTabelaLivros
CodeIgniter v4.6.1 Command Line Tool - Server Time: 2025-07-10 11:35:22 UTC+00:00
File created: APPPATH/Database/Migrations/2025-07-10-113522_CriarTabelaLivros.php
```

Veja que um novo arquivo foi criado



```
2025-07-10-113522_CriarTabelaLivros.php ×  
app > Database > Migrations > 2025-07-10-113522_CriarTabelaLivros.php > ...  
1 <?php  
2  
3 namespace App\Database\Migrations;  
4  
5 use CodeIgniter\Database\Migration;  
6  
7 class CriarTabelaLivros extends Migration  
8 {  
9     public function up()  
10    {  
11        //  
12    }  
13  
14    public function down()  
15    {  
16        //  
17    }  
18 }  
19
```

2025-07-10-113522_CriarTabelaLivros.php ×

app > Database > Migrations > 2025-07-10-113522_CriarTabelaLivros.php > ...

```
1  <?php
2
3  namespace App\Database\Migrations;
4
5  use CodeIgniter\Database\Migration;
6
7  class CriarTabelaLivros extends Migration
8  {
9      public function up()
10     {
11         $this->forge->addField('id');
12         $this->forge->addField([
13             'titulo' => ['type' => 'VARCHAR', 'constraint' => 255],
14             'autor' => ['type' => 'VARCHAR', 'constraint' => 255],
15             'created_at' => ['type' => 'DATETIME', 'null' => TRUE],
16             'updated_at' => ['type' => 'DATETIME', 'null' => TRUE],
17             'deleted_at' => ['type' => 'DATETIME', 'null' => TRUE],
18         )->createTable('livros');
19     }
20
21     public function down()
22     {
23         $this->forge->dropTable('livros');
24     }
25 }
```

Adicione as colunas na migrate

Execute a migrate

No terminal, execute o comando:

```
● (base) diego@hipogrifo-341 livros % php spark migrate

CodeIgniter v4.6.1 Command Line Tool - Server Time: 2025-07-10 11:42:18 UTC+00:00

Running all new migrations...
    Running: (App) 2025-07-10-113522_App\Database\Migrations\CriarTabelaLivros
Migrations complete.
```

Veja que um novo arquivo
foi criado em
/writable/db



O arquivo não é exibido no editor de texto porque é binário ou usa
uma codificação de texto não suportada.

[Abrir Mesmo Assim](#)

Não é possível ler o arquivo

Cada model é uma tabela do banco

Criando modelos (Models)

Models são utilizados para manipular os dados.

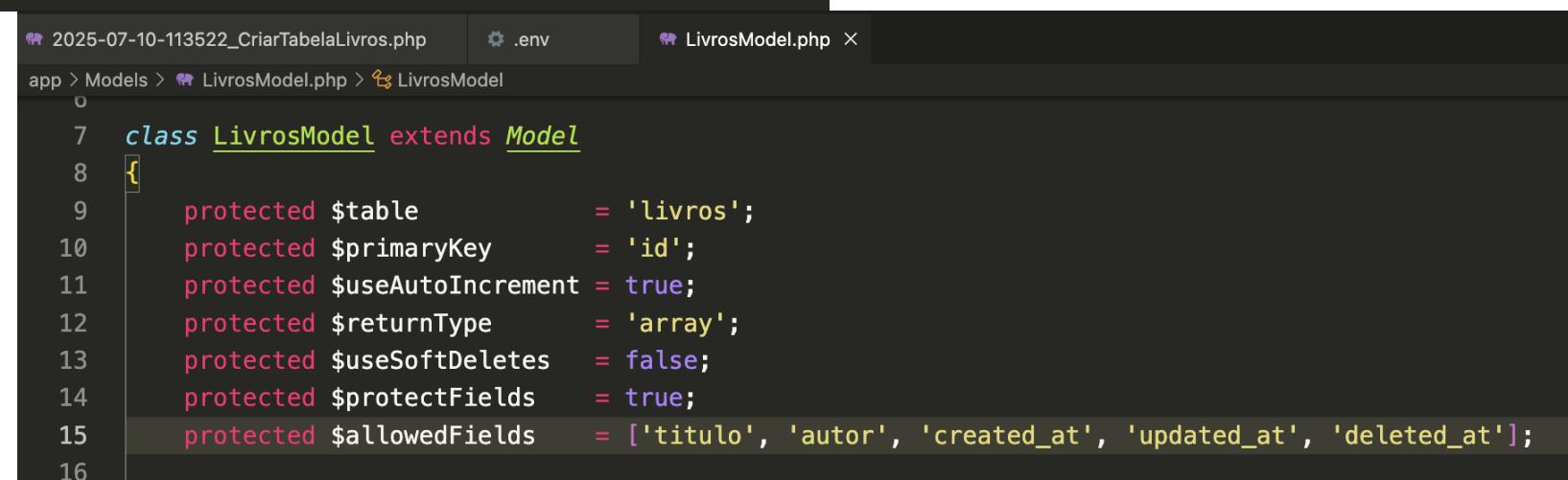
Para criar um modelo:

```
php spark make:model LivrosModel
```

```
● (base) diego@hipogrifo-341 livros % php spark make:model LivrosModel

CodeIgniter v4.6.1 Command Line Tool – Server Time: 2025-07-10 11:45:23 UTC+00:00

File created: APPPATH/Models/LivrosModel.php
```



The screenshot shows a terminal window with the following output:

```
● (base) diego@hipogrifo-341 livros % php spark make:model LivrosModel

CodeIgniter v4.6.1 Command Line Tool – Server Time: 2025-07-10 11:45:23 UTC+00:00

File created: APPPATH/Models/LivrosModel.php
```

Below the terminal, a code editor window displays the generated `LivrosModel.php` file. The file is located at `app/Models/LivrosModel.php`. The code defines a new model class `LivrosModel` that extends the `Model` class. It includes properties for the table name (`$table`), primary key (`$primaryKey`), auto-increment behavior (`$useAutoIncrement`), return type (`$returnType`), soft deletes (`$useSoftDeletes`), protected fields (`$protectFields`), and allowed fields (`$allowedFields`). The code editor interface shows tabs for `2025-07-10-113522_CriarTabelaLivros.php`, `.env`, and `LivrosModel.php`.

```
2025-07-10-113522_CriarTabelaLivros.php .env LivrosModel.php
app > Models > LivrosModel.php > LivrosModel
o
7  class LivrosModel extends Model
8  {
9      protected $table      = 'livros';
10     protected $primaryKey   = 'id';
11     protected $useAutoIncrement = true;
12     protected $returnType    = 'array';
13     protected $useSoftDeletes = false;
14     protected $protectFields = true;
15     protected $allowedFields = ['titulo', 'autor', 'created_at', 'updated_at', 'deleted_at'];
16
```

Modifique o controller

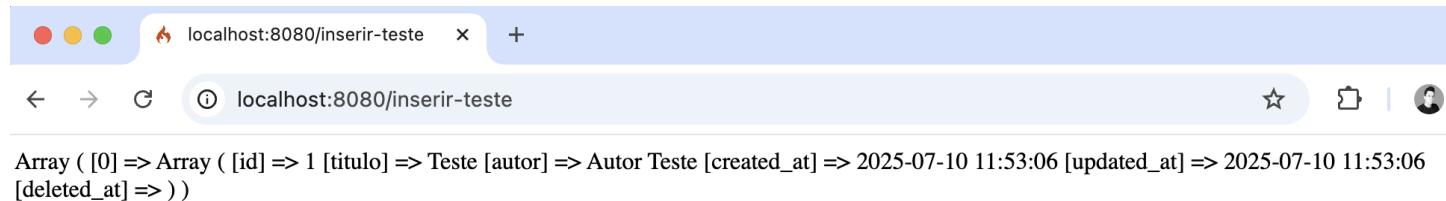
Povoando o banco de dados

Crie uma nova rota



```
Livros.php x Routes.php x
app > Config > Routes.php
9
10 $routes->get('/livros', 'Livros::listar');
11 $routes->get('/inserir-teste', 'Livros::inserir teste');
```

Acesse o endereço da rota



```
localhost:8080/inserir-teste
localhost:8080/inserir-teste
Array ( [0] => Array ( [id] => 1 [titulo] => Teste [autor] => Autor Teste [created_at] => 2025-07-10 11:53:06 [updated_at] => 2025-07-10 11:53:06 [deleted_at] => ) )
```

Livros.php x

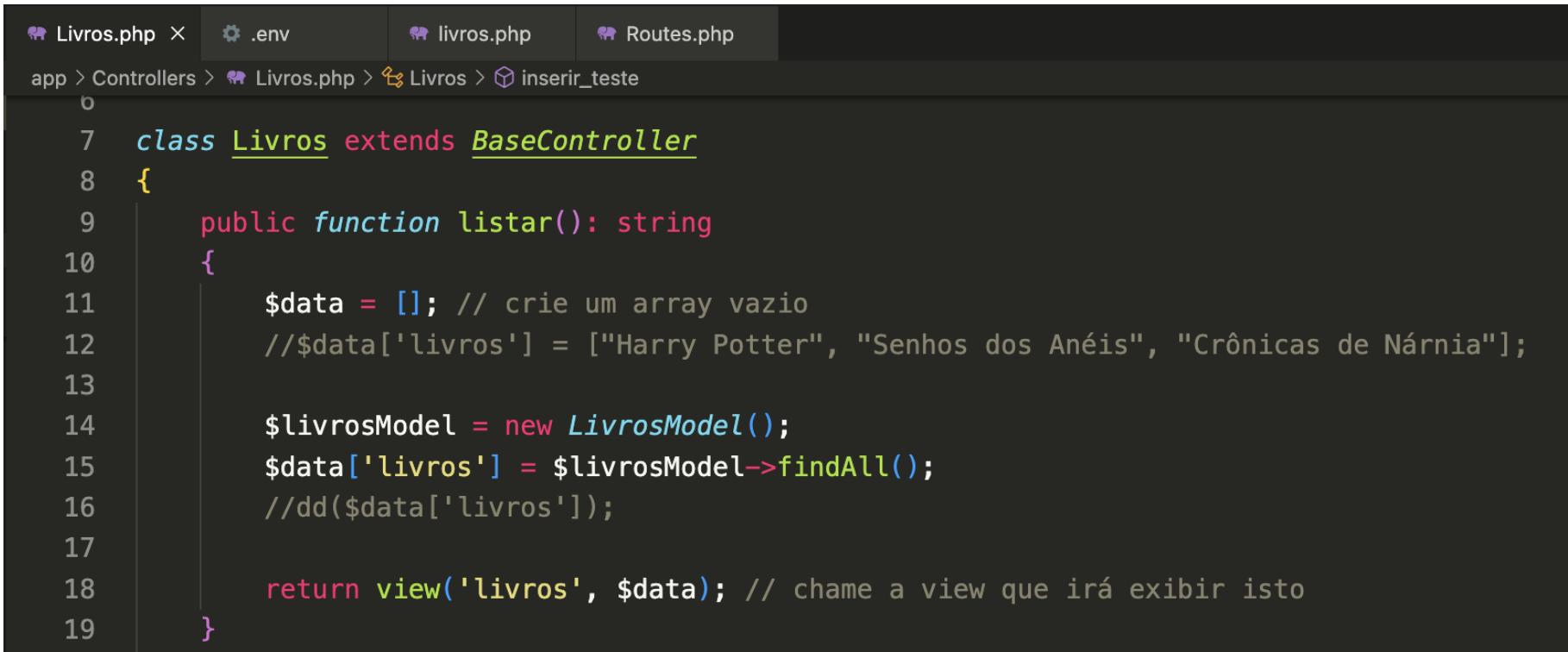
```
app > Controllers > Livros.php > {} App\Controllers
1 <?php
2
3 namespace App\Controllers;
4 use App\Models\LivrosModel;
5 use CodeIgniter\I18n\Time;
```

```
public function inserir_teste()
{
    $livrosModel = new LivrosModel();
    $agora = new Time('now');

    $livrosModel->save([
        'titulo'=>'Teste',
        'autor'=>'Autor Teste',
        'created_at'=>$agora,
        'updated_at'=>$agora
    ]);

    print_r($livrosModel->findAll());
}
```

Modificando o controller Livros



The screenshot shows a code editor with the file `Livros.php` open. The file contains PHP code for a controller named `Livros`. The code defines a method `listar()` that retrieves all books from a database and returns them to a view. The code editor also shows the project structure at the top, including files like `.env`, `livros.php`, and `Routes.php`.

```
Livros.php × .env livros.php Routes.php
app > Controllers > Livros.php > Livros > inserir_teste
  6
  7 class Livros extends BaseController
  8 {
  9     public function listar(): string
 10    {
 11        $data = []; // crie um array vazio
 12        //$data['livros'] = ["Harry Potter", "Senhos dos Anéis", "Crônicas de Nárnia"];
 13
 14        $livrosModel = new LivrosModel();
 15        $data['livros'] = $livrosModel->findAll();
 16        //dd($data['livros']);
 17
 18        return view('livros', $data); // chame a view que irá exibir isto
 19    }
}
```

Livros.php

.env

livros.php X

Routes.php

app > Views > livros.php > ...

```
1  <?= $this->extend('template') ?>
2  <?= $this->section('conteudo') ?>
3
4  <h1>Lista de livros</h1>
5  <table class="table table-hover table-striped">
6      <thead>
7          <tr>
8              <th>Livro</th>
9              <th>Autor</th>
10         </tr>
11     </thead>
12     <tbody>
13         <?php foreach($livros as $livro){ ?>
14             <tr>
15                 <td><?= $livro['titulo'] ?></td>
16                 <td><?= $livro['autor'] ?></td>
17             </tr>
18         <?php } // fim foreach ?>
19     </tbody>
20 </table>
21
22 <?= $this->endSection() ?>
23
```

View livros

Livros

localhost:8080/livros

Lista de livros

Livro	Autor
Teste	Autor Teste

©2025 Livros - Todos os direitos reservados.

Adicione uma nova rota (Routes.php)

```
Livros.php .env LivrosModel.php livros.php Routes.php X
app > Config > Routes.php
5 /**
8 $routes->get('/', 'Home::index');
9
10 $routes->get('/livros', 'Livros::listar');
11 $routes->get('/inserir-teste', 'Livros::inserir_teste');
13 $routes->get('/deletar/(:num)', 'Livros::deletar/$1');
```

Crie a função deletar no Controller

```
Livros.php .env LivrosModel.php livros.php Routes.php ...
app > Controllers > Livros.php > ...
7 class Livros extends BaseController
36
37     public function deletar($id){
38         /* Remove um livro com base no ID fornecido */
39
40         $livrosModel = new LivrosModel();
41         $livrosModel->delete(['id' => $id]);
42
43         $data = []; // crie um array vazio
44         $data['deletado'] = $id;
45
46         return redirect('livros')->with('mensagem', "Livro $id deletado com sucesso");
47     }
48 }
```

```
Livros.php .env LivrosModel.php livros.php X Routes.php
app > Views > livros.php > table.table-table-hover.table-striped > tbody > tr > td
4 <h1>Lista de livros</h1>
5 <table class="table table-hover table-striped">
6     <thead>
7         <tr>
8             <th>Livro</th>
9             <th>Autor</th>
10            <th>Deletar</th>
11        </tr>
12    </thead>
13    <tbody>
14        <?php foreach($livros as $livro){ ?>
15        <tr>
16            <td><?= $livro['titulo'] ?></td>
17            <td><?= $livro['autor'] ?></td>
18            <td>
19                <a href=<?=base_url('/deletar/'.$livro['id'])?>>">
20                    <i class="bi bi-trash3-fill"></i>
21                </a>
22            </td>
23        </tr>
24    <?php } // fim foreach ?>
25    </tbody>
26 </table>
27
```

Crie uma nova coluna na tabela

Livro	Autor	Deletar
Teste	Autor Teste	

Adicionando uma mensagem (view livros.php)

```
4 <?php if (session()->has('mensagem')): ?>
5   <div class="alert alert-danger">
6     <?= session('mensagem') ?>
7   </div>
8 <?php endif; ?>
```

Adicione mais livros e teste a deleção

A screenshot of a web browser window titled "Livros". The address bar shows "localhost:8080/index.php/livros". The main content area displays a success message "Livro 17 deletado com sucesso" in a pink box. Below it, a table titled "Lista de livros" lists six books, each with a delete icon.

Livro	Autor	Deletar
Teste	Autor Teste	trash

<http://localhost:8080/inserir-teste>

Tarefa de casa

- Construa páginas para:
 - Inserir novos dados
 - Editar dados

Como colocar no ar

The screenshot shows the GitHub Desktop application interface. At the top, there are three tabs: 'Current Repository' (cocada-web), 'Current Branch' (main), and 'Fetch origin' (Last fetched 20 minutes ago). A notification bar at the top left indicates an updated version of GitHub Desktop is available.

In the center, a modal window titled 'Create a New Repository' is open. It contains fields for 'Name' (livros), 'Description' (empty), 'Local Path' (/Users/diego/Documents/GitHub), and 'Git Ignore' (CodeIgniter). There is also a checkbox for 'Initialize this repository with a README'. Below these fields, a note states: 'The repository will be created at /Users/diego/Documents/GitHub/livros.' At the bottom of this modal are 'Cancel' and 'Create Repository' buttons.

To the right of the main window, a tooltip provides instructions: 'Publish your repository to GitHub' and 'Always available in the toolbar for local repositories or ⌘ P'. It includes a 'Publish repository' button.

At the bottom right, another modal window titled 'Publish Repository' is open, specifically for 'GitHub.com'. It has fields for 'Name' (livros), 'Description' (empty), and a checkbox for 'Keep this code private'. Below these are 'Organization' (None) and 'GitHub Enterprise' tabs. At the bottom are 'Cancel' and 'Publish Repository' buttons.

- Cole a pasta do projeto dentro do diretório do GitHub

Pronto!

The screenshot shows a browser window with two tabs: 'localhost:8080/inserir-teste' and 'GitHub - dcbmariano/livros'. The GitHub tab is active, displaying the repository 'dcbmariano/livros' which is public. The repository has 1 branch and 0 tags. The code tab is selected, showing a single commit from 'dcbmariano' titled 'Initial commit' made 2 minutes ago. The commit details show it was an initial commit for files like app, public, tests, writable, .env, .gitattributes, and .gitignore. The repository page also includes sections for About (no description, website, or topics provided), Readme, MIT license, Activity, 0 stars, 0 watching, 0 forks, and a Report repository button. Below the repository details, there's a Releases section stating 'No releases published'.

localhost:8080/inserir-teste

GitHub - dcbmariano/livros

github.com/dcbmariano/livros

Product Solutions Resources Open Source Enterprise Pricing

dcbmariano / livros Public

Code Issues Pull requests Actions Projects Security Insights

main 1 Branch 0 Tags

Go to file

Code

dcbmariano Initial commit · 0690349 · 2 minutes ago 1 Commit

File	Commit Message	Time Ago
app	Initial commit	2 minutes ago
public	Initial commit	2 minutes ago
tests	Initial commit	2 minutes ago
writable	Initial commit	2 minutes ago
.env	Initial commit	2 minutes ago
.gitattributes	Initial commit	2 minutes ago
.gitignore	Initial commit	2 minutes ago

About

No description, website, or topics provided.

Readme

MIT license

Activity

0 stars

0 watching

0 forks

Report repository

Releases

No releases published

Colocando em produção

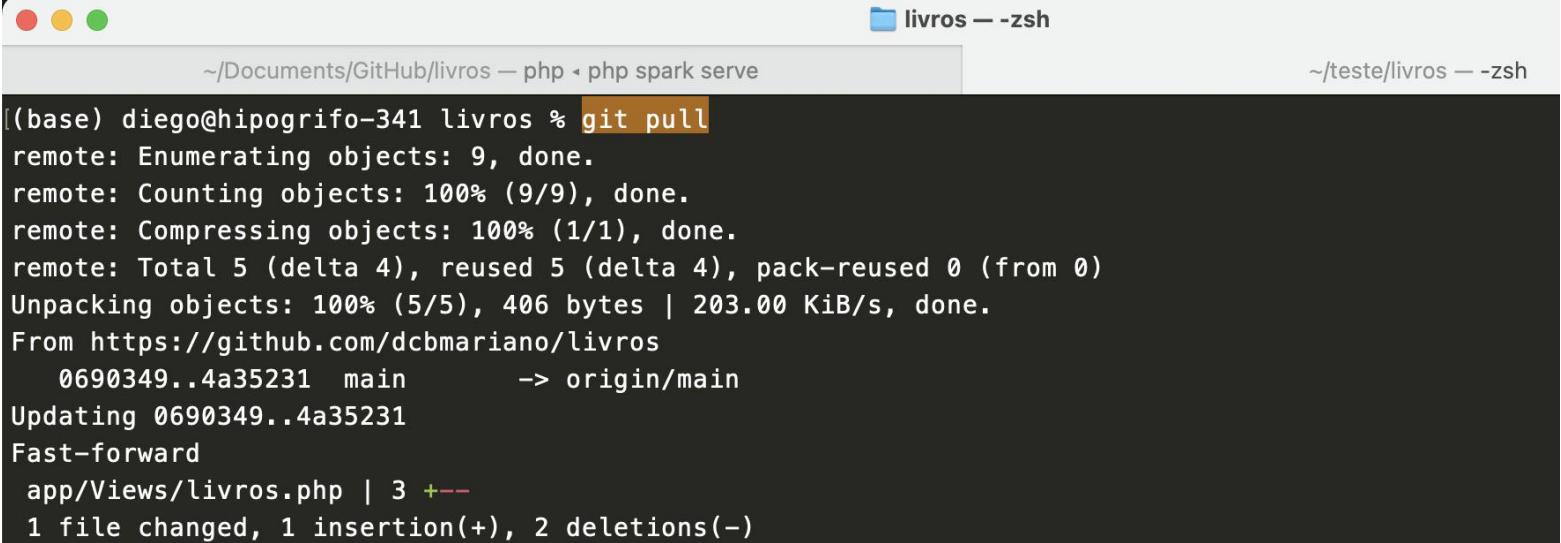
Clone o repositório

```
(base) diego@hipogrifo-341 teste % git clone https://github.com/dcbmariano/livros.git
Cloning into 'livros'...
remote: Enumerating objects: 145, done.
remote: Counting objects: 100% (145/145), done.
remote: Compressing objects: 100% (114/114), done.
Receiving objects: 100% (145/145), 104.39 KiB | 2.22 MiB/s, done.
remote: Total 145 (delta 21), reused 145 (delta 21), pack-reused 0 (from 0)
Resolving deltas: 100% (21/21), done.
```

Edite o .env

```
23 | #app.baseURL = 'http://localhost:8080'
24 | app.baseURL = 'https://www.seu-site.com'
```

Atualize a versão do servidor

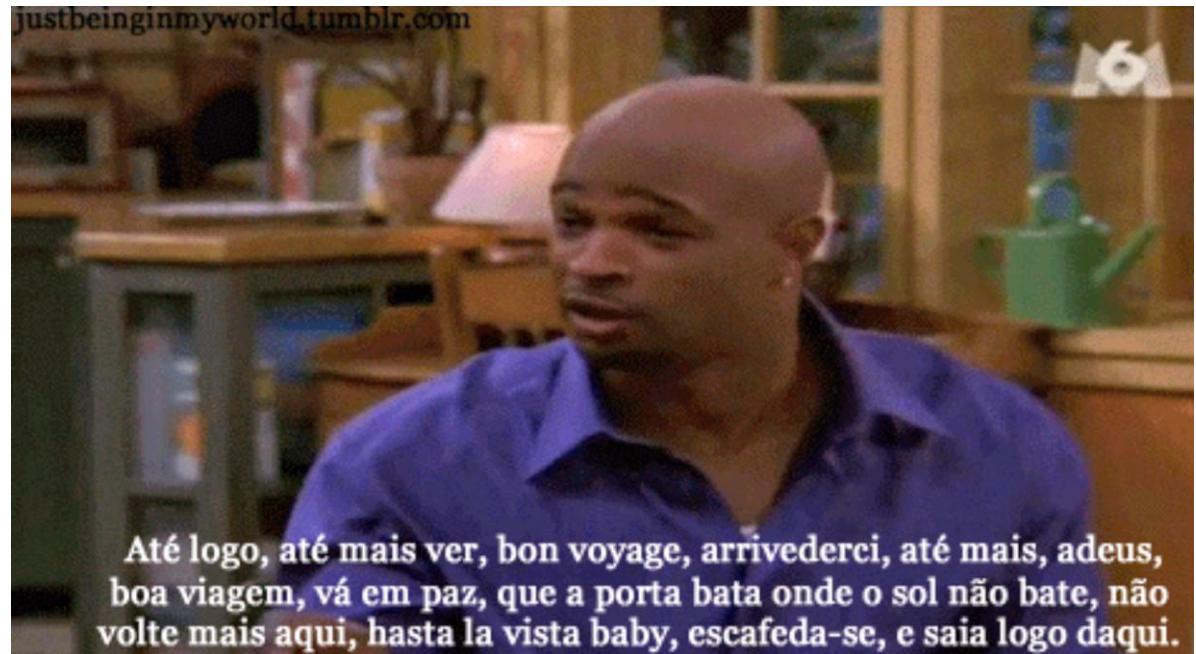


The screenshot shows a Mac OS X terminal window titled "livros — zsh". The window has three tabs: the active tab is labeled "~/Documents/GitHub/livros — php < php spark serve", and two other tabs are labeled "~/teste/livros — zsh". The terminal output shows the execution of a Git pull command:

```
(base) diego@hipogrifo-341 livros % git pull
remote: Enumerating objects: 9, done.
remote: Counting objects: 100% (9/9), done.
remote: Compressing objects: 100% (1/1), done.
remote: Total 5 (delta 4), reused 5 (delta 4), pack-reused 0 (from 0)
Unpacking objects: 100% (5/5), 406 bytes | 203.00 KiB/s, done.
From https://github.com/dcbmariano/livros
  0690349..4a35231 main      -> origin/main
Updating 0690349..4a35231
Fast-forward
  app/Views/livros.php | 3 +--
  1 file changed, 1 insertion(+), 2 deletions(-)
```

Obrigado!

- Fim
- *The end*
- Beijo na mucosa do pulmão de vocês!



Até logo, até mais ver, bon voyage, arrivederci, até mais, adeus,
boa viagem, vá em paz, que a porta bata onde o sol não bate, não
volte mais aqui, hasta la vista baby, escafeda-se, e saia logo daqui.