

## Custom Post Type, listagem de imóveis e single.php

### Transcrição

Anteriormente, vimos como implementar o `cadastro_post_type_imoveis()`, customizado justamente para representar nossos imóveis de forma coerente para quem for baixar e utilizar nosso tema futuramente. Também alteramos o ícone localizado ao lado do link de "Imóveis", as *labels*, de "Editar post" para "Editar imóvel", melhorando o dashboard cada vez mais.

Continuando, se acessarmos a home da nossa página teremos, atualmente, três imóveis: casa da praia, casa de argila e casa de madeira. A questão é: eles são posts ou imóveis? Para averiguarmos isto, vamos à "Posts" e deletar todos os que ali estão, e com isso devemos informar ao Wordpress que não são os posts que deverão ser trazidos do banco de dados, e sim os imóveis.

Criaremos um novo objeto do Wordpress em `index.php`, onde se encontra o loop, chamado `WP_Query()`, responsável pelas buscas, as queries do loop. Além disso, passaremos alguns argumentos, e salvaremos a variável no loop. Assim, o `$loop` filtrará pelos imóveis, e `have_posts()` precisará se alinhar a isso, portanto acrescentaremos `$loop->` antes dele, de modo que um método seu seja acessado.

```
<?php

$args = array( 'post_type' => 'imovel' );
$loop = new WP_Query( $args );
if( $loop->have_posts() ) { ?>
<ul class="imoveis-listagem">
<?php while( $loop->have_posts() ) {
    $loop->the_post();
?>

<li class="imoveis-listagem-item">
<?php the_post_thumbnail(); ?>
<h2><?php the_title(); ?></h2>
<div><?php the_content(); ?></div>

</li>
<?php } ?>
```

Ao retornarmos ao nosso site, tudo continuará funcionando conforme esperado. Pensando no que o usuário faria, ao se interessar por uma das casas, provavelmente ele tentaria clicar na imagem correspondente para acessar maiores informações. No entanto, não temos nenhum link; no dashboard, ao clicarmos em "Imóveis" e escolhermos uma delas, teremos um "Link permanente" gerado automaticamente pelo Wordpress.

Se o acessarmos, porém, seremos redirecionados ao `index`. Isto porque a página do imóvel ainda não foi criada. Faremos isso criando um novo arquivo, `single.php`, seguindo uma nomenclatura padrão do Wordpress para agrupar os *post type custom* feitos. Desta vez, se acessarmos o link permanente, teremos uma página vazia, em branco, o que era de se esperar, já que ainda não incluímos nada nela.

Por padrão, precisaremos colocar um `get_header()` para que o cabeçalho seja pego, e um `get_footer()` para o rodapé:

```
<?php get_header(); ?>
<?php get_footer(); ?>
```

Testaremos a página no navegador e, confirmado seu bom funcionamento, acrescentaremos mais código em `single.php`. No miolo, será necessário solicitarmos a renderização de algo, usando-se o loop, isto é, precisaremos listar os imóveis novamente.

```
<?php get_header(); ?>

<?php
if( have_posts() ) {
    while( have_posts() ) {
        the_post();

    }

    <?php the_post_thumbnail(); ?>
    <?php the_title(); ?>
    <?php the_content(); ?>

    <?php
    }
}

<?php get_footer(); ?>
```

Voltaremos à página e a atualizaremos, para verificar como ela é renderizada. A imagem aparece em tamanho máximo, seguida do título e descrição. Retornando ao código de `single.php`, envolveremos todo seu conteúdo em tags `<main>` e `<article>`. E para podermos incluir o CSS, acrescentaremos código HTML.

```
<?php get_header(); ?>

<main>

<article>

    <?php
    if( have_posts() ) {
        while( have_posts() ) {
            the_post();

        }

        <div class="single-imovel-thumbnail">
            <?php the_post_thumbnail(); ?>
        </div>

        <div class="container">
            <section class="chamada-principal">
                <?php the_title(); ?>
            </section>

            <section class="single-imovel-geral">
                <div class="single-imovel-descricao">
```

```

        <?php the_content(); ?>
    </div>
</section>

</div>

<?php
}
}

?>
</article>
</main>

<?php get_footer(); ?>

```

Com isso, atualizaremos a página, porém não teremos nenhuma mudança visível — a partir de agora, teremos que modificar `style.css`, incluindo um trecho de código preparado previamente para isso, bastando copiar e colá-lo no arquivo. Isso fará com que a imagem fique ajustada proporcionalmente, acompanhada do título e descrição, também estilizados.

Poderemos incluir mais informações, como a data de criação, e para isto abriremos `single.php` e, ainda dentro da `<div>` de classe `container`, acrescentaremos o seguinte trecho:

```

<span class="single-imovel-data">
    <?php the_date(); ?>
</span>

```

Voltaremos à nossa página para verificar o resultado. Voltaremos à home para lidar com aquela questão da linkagem à página específica do imóvel. Para incluirmos um link na imagem exibida na página inicial, acrescentaremos uma tag `<a>` em `index.php`. Porém, os links dos imóveis devem ser gerados automaticamente, portanto utilizaremos a função `the_permalink()`, do Wordpress.

```

<li class="imoveis-listagem-item">
    <a href="<?php the_permalink(); ?>">
        <?php the_post_thumbnail(); ?>
        <?php the_title(); ?>
        <?php the_content(); ?>
    </a>
</li>

```

Tudo funcionará bem! Aprendemos a listar postagens, ou seja, imóveis, em nosso loop, vimos como criar uma página que renderize e mostre nossos imóveis, com `single.php`, em que importamos cabeçalho e rodapé, implementamos um pouco de HTML. No loop, apenas um post (imóvel) foi pego, justamente aquele que for clicado na página inicial.