模式

目录
区块模式
register_block_pattern
注销区块模式
unregister_block_pattern
区块模式分类
register_block_pattern_category unregister_block_pattern_category
与区块类型和模式转换相关的区块模式
语义区块模式

区块模式是从区块插入器的模式选项卡中可用的预定义区块布局。一旦插入到内容中，这些区块即可用于添加
或修改内容及配置。

区块模式

register_block_pattern
编辑器内置了若干核心区块模式。主题和插件开发者可以使用 register_block_pattern 辅助函数注册额外的自定义区块模式。

register_block_pattern 辅助函数接收两个参数：

title：遵循 命名空间/标题 命名约定的机器可读标题。
properties：描述模式属性的数组。

区块模式可用的属性包括：

title（必需）：模式的人类可读标题。
content （必需）：模式的区块 HTML 标记。 description
（可选）：在插入器中用于描述模式的视觉隐藏文本。描述是可选的，但当标题未能完全描述模式功

能时，强烈建议提供。描述将帮助用户在搜索时发现该模式。

- `categories` （可选）：用于对区块模式进行分组的已注册模式类别数组。区块模式可以显示在多个类别中。类别必须单独注册才能在此处使用。
- `keywords` （可选）：帮助用户在搜索时发现模式的别名或关键词数组。
- `viewportWidth` （可选）：指定模式预期宽度的整数，以便在插入器中显示模式的缩放预览。
- `blockTypes` （可选）：模式预期与之配合使用的区块类型数组。每个值都必须是已声明区块的 `name` 。
- `postTypes` （可选）：模式被限制只能与之配合使用的文章类型数组。该模式仅在编辑数组中传递的任一文章类型时可用。对于所有其他文章类型，该模式完全不可用。
- `templateTypes` （可选）：模式适用的模板类型数组，例如，如果模式用于 404 页面则为 `404` ，如果模式用于显示单篇文章则为 `single-post` 。
- `inserter` （可选）：默认情况下，所有模式都会出现在插入器中。要隐藏某个模式，使其只能通过编程方式插入，请将 `inserter` 设置为 `false` 。
- `source` （可选）：表示模式来源的字符串。对于插件注册的模式，传递字符串 `plugin` 。对于主题，传递字符串 `theme` 。

以下代码示例注册了一个名为 `my-plugin/my-awesome-pattern` 的区块模式：

```
register_block_pattern(
□'my-plugin/my-awesome-pattern',
□array(
□□'title'       => __( 'Two buttons', 'my-plugin' ),
□□'description' => _x( 'Two horizontal buttons, the left button is filled in, and the right button is outlined.', 'Block pattern description', 'my-plugin' ),
□□'content'     => "<!-- wp:buttons {\"align\":\"center\"} -->\n<div class=\"wp-block-buttons aligncenter\"><!-- wp:button {\"backgroundColor\":\"very-dark-gray\",\"borderRadius\":0} -->\n<div class=\"wp-block-button\"><a class=\"wp-block-button__link has-background has-very-dark-gray-background-color no-border-radius\">" . esc_html__( 'Button One', 'my-plugin' ) . "</a></div>\n<!-- /wp:button -->\n\n<!-- wp:button {\"textColor\":\"very-dark-gray\",\"borderRadius\":0,\"className\":\"is-style-outline\"} -->\n<div class=\"wp-block-button is-style-outline\"><a class=\"wp-block-button__link has-text-color has-very-dark-gray-color no-border-radius\">" . esc_html__( 'Button Two', 'my-plugin' ) . "</a></div>\n<!-- /wp:button --></div>\n<!-- /wp:buttons -->",
□)
);
```

Note that `register_block_pattern()` should be called from a handler attached to the `init` hook.

```
function my_plugin_register_my_patterns() {
  register_block_pattern( ... );
}

add_action( 'init', 'my_plugin_register_my_patterns' );
```

注销区块模式

unregister_block_pattern

`unregister_block_pattern` 辅助函数允许从主题或插件中注销先前注册的区块模式，它接收一个参数。

- `title`：要注销的区块模式的名称。

以下代码示例注销了名为 `my-plugin/my-awesome-pattern` 的区块模式：

```
unregister_block_pattern( 'my-plugin/my-awesome-pattern' );
```

注意：

`unregister_block_pattern()` 应在附加到 init 钩子的处理程序中调用。

```
function my_plugin_unregister_my_patterns() {
  unregister_block_pattern( ... );
}

add_action( 'init', 'my_plugin_unregister_my_patterns' );
```

区块模式分类

区块模式可通过分类进行分组。区块编辑器自带预设分类，可用于自定义区块模式。您也可以注册自己的区块模式分类。

register_block_pattern_category

`register_block_pattern_category` 辅助函数接收两个参数。

- `title`：区块模式分类的机器可读标题。
- `properties`：描述模式分类属性的数组。

模式分类的属性包括：

- `label`（必需）：模式分类的人类可读标签。

以下代码示例注册了名为 `hero` 的分类：

```
register_block_pattern_category(
'hero',
array( 'label' => __( 'Hero', 'my-plugin' ) )
);
```

注意：

`register_block_pattern_category()` 应从附加到 init 钩子的处理程序中调用。

除非有模式被分配到该分类，否则该分类不会显示在 “模式 ”下。

```
function my_plugin_register_my_pattern_categories() {
  register_block_pattern_category( ... );
}

add_action( 'init', 'my_plugin_register_my_pattern_categories' );
```

unregister_block_pattern_category

`unregister_block_pattern_category`
辅助函数允许从主题或插件中注销先前注册的区块模式类别，并接收一个参数。

- `title`：要注销的区块模式类别的名称。

以下代码示例注销名为 `hero` 的类别：

```
unregister_block_pattern_category( 'hero' );
```

注意：

`unregister_block_pattern_category()` 应在附加到 init 钩子的处理程序中调用。

```
function my_plugin_unregister_my_pattern_categories() {
  unregister_block_pattern_category( ... );
}

add_action( 'init', 'my_plugin_unregister_my_pattern_categories' );
```

与区块类型和模式转换相关的区块模式

可以将区块模式附加到一个或多个区块类型上。这会将该区块模式添加为该区块类型的可用转换选项。

目前，这些转换仅适用于简单区块（没有内部区块的区块）。要使模式被建议，
每个选中的区块都必须存在于区块模式中 。

例如：

```
register_block_pattern(
□'my-plugin/powered-by-wordpress',
□array(
□□'title'      => __( 'Powered by WordPress', 'my-plugin' ),
□□'blockTypes' => array( 'core/paragraph' ),
□□'content'    => '<!-- wp:paragraph {"backgroundColor":"black","textColor":"white"} -->
□□<p class="has-white-color has-black-background-color has-text-color has-background">Powered b
WordPress</p>
□□<!-- /wp:paragraph -->',
□)
);
```

上面的代码注册了一个名为 `my-plugin/powered-by-wordpress` 的区块模式，并在段落区块的 “转换菜单 ”
中显示该模式。转换结果将保留段落的现有内容，并应用其他属性 ——在本例中是背景色和文本颜色。

如上所述，如果我们选择了多个区块，并且存在与这些区块匹配的上下文模式，那么简单区块的模式转换也可
以工作。让我们看一个附加了两个区块类型的模式示例。

```
register_block_pattern(
□'my-plugin/powered-by-wordpress',
□array(
□□'title'      => __( 'Powered by WordPress', 'my-plugin' ),
□□'blockTypes' => array( 'core/paragraph', 'core/heading' ),
□□'content'    => '<!-- wp:group -->
□□□□□□<div class="wp-block-group">
```

```
□□□□□□□<!-- wp:heading {"fontSize":"large"} -->
□□□□□□□<h2 class="has-large-font-size"><span style="color:#ba0c49" class="has-inline-color">Hi everyone</span></h2>
□□□□□□□<!-- /wp:heading -->
□□□□□□□<!-- wp:paragraph {"backgroundColor":"black","textColor":"white"} -->
□□□□□□□<p class="has-white-color has-black-background-color has-text-color has-background">Power WordPress</p>
□□□□□□□<!-- /wp:paragraph -->
□□□□□□□</div><!-- /wp:group -->',
□)
);
```

在上面的例子中，如果我们选择 两个区块类型中的一个，无论是段落还是标题区块，这个模式都会被建议，通过使用其内容转换选中的区块，并添加模式中剩余的区块。另一方面，如果我们多选一个段落和一个标题区块，两个区块都将被转换。

区块也可以在其他地方使用这些上下文区块模式。例如，当插入一个新的查询循环区块时，会向用户提供附加到该区块的所有模式列表。

语义区块模式

在区块主题中，您还可以将区块模式标记为 "页眉 "或 "页脚 "模式（模板部件区域）。我们称这些为 " 语义区块模式 "。当插入或替换页眉 /页脚模板部件时，这些模式会展示给用户。

示例:

```
<?php
register_block_pattern(
□'my-plugin/my-header',
□array(
□□'title'      => __(我的页眉', 'my-plugin' ),
□□'categories' => array( 'header' ),
□□//该模式分配"header"区域
□□'blockTypes' => array( 'core/template-part/header' ),
□□'content'    =>我的区块模式内容',
□)
);
```

版本号 #2
由 壳壳虫 创建于 2026-03-26 17:08:43 UTC
由 壳壳虫 更新于 2026-03-26 17:08:43 UTC