

## gatsby-remark-custom-blocks

**Note:** this plugin is incompatible with `gatsby-transformer-remark@4.0.0` because the upstream `remark-custom-blocks` package is not upgraded to remark 13 yet.

The work on upgrading to remark 13 is in progress, so follow this [issue](#) for updates.

The latest compatible version is `gatsby-transformer-remark@3.2.0`.

Adds custom blocks to `MarkdownRemark` using `remark-custom-blocks`.

Unlike in `gatsby-remark-component` where you can only use HTML within the custom component tag, custom blocks allow you to use markdown within the block.

### Install

```
npm install gatsby-remark-custom-blocks
```

### How to use

The configuration object for custom blocks follows this specification:

```
trigger: {
  classes: String, space-separated classes, optional, default: ''
  title: String, 'optional' | 'required', optional, default: custom titles not allowed
  details: boolean, optional, default: false
}
```

Here is an example of how you can implement custom blocks in your `gatsby-config.js` file:

```
plugins: [
  {
    resolve: `gatsby-transformer-remark`,
    options: {
      plugins: [
        {
          resolve: "gatsby-remark-custom-blocks",
          options: {
            blocks: {
              danger: {
                classes: "danger",
              },
              info: {
                classes: "info",
                title: "optional",
              },
            },
          },
        },
      ],
    },
  },
]
```

```

    },
  },
},
],
},
},
]

```

Use the following Markdown syntax to create blocks in your file:

```

[[danger]]
| content

```

```

[[info | This is a title!]]
| content

```

This will generate the following html:

```

<div class="custom-block danger">
  <div class="custom-block-body"><p>content</p></div>
</div>

<div class="custom-block info">
  <div class="custom-block-heading">This is a title!</div>
  <div class="custom-block-body"><p>content</p></div>
</div>

```

The `title` configuration option can be passed either `optional` or `required`. When the required option is set in your `gatsby-config.js` the custom block will not be rendered if there is not a title argument in your markdown.

If you set the `details` configuration option to `true` the outer `<div>` will be changed to a details tag. If you have a title set, the title will be used as the summary.

Here is an example configuration for using details:

```

blocks: {
  danger: {
    classes: "danger",
    details: true,
  },
  info: {
    title: "optional",
    details: true,
  },
},

```

And here is the html output considering the same markdown as before:

```
<details class="custom-block danger">
  <div class="custom-block-body"><p>content</p></div>
</details>

<details class="custom-block">
  <summary class="custom-block-heading">This is a title!</summary>
  <div class="custom-block-body"><p>content</p></div>
</details>
```