

The Admonitions Extension

markdown.extensions.admonition

*This is a quick reference for VSCode's **admonition extension**, which allows to display messages in fancy and vibrant colored boxes and use icon + text headers to grab readers' attention. This extension is included in the standard Markdown library. The full documentation is available on [Markdown's Officially Supported Extension's Website](#).*

▼ TABLE OF CONTENTS

- [Admonitions Overview](#)
- [Creating Admonitions](#)
- [Default Types Of Admonitions](#)
- [Customizing Admonitions](#)
 - [Transpiled HTML](#)
 - [Custom Types Of Admonitions](#)
 - [Customizing Admonitions' Styles](#)

Admonitions Overview

The typical use case for *admonitions* is to draw the reader's attention to a particular point. They can appear anywhere in an ordinary HTML `body` element, including another *admonition*.

Admonitions consist of a colored box with an optional header and a body. The header is composed of an icon and a title, both of which can be customized. The body is a normal paragraph of text.

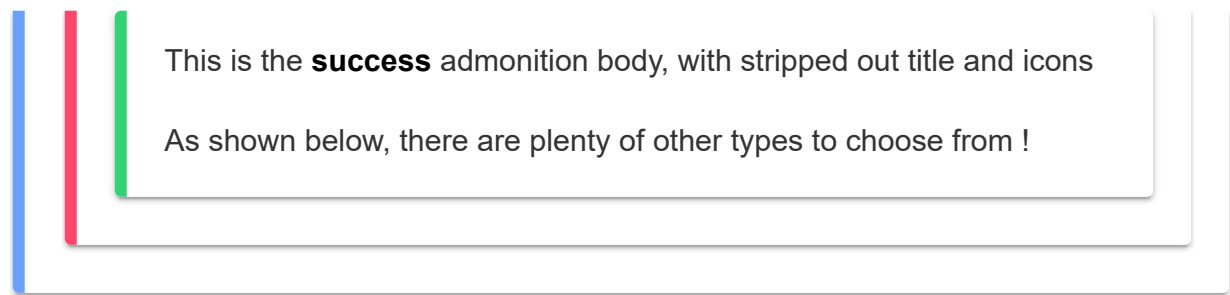
Below is a quick example of multiple nested admonitions of different types:

Note

This is the **note** admonition body

Danger Title

This is the **danger** admonition body



Creating Admonitions

The extension `markdown.extensions.admonition` provides the following syntax for admonitions:

```
!!! <type> [Alternative Title | "" to hide the header]

<paragraph>

<paragraph>

...
```

, where the `<type>` defines the overall style (icon and color) of the admonition and the `<paragraph>` is a normal paragraph of text. As an example, here is the code that generated the admonitions of the *overview* section:

```
!!! note

    This is the note admonition body

!!! danger Danger Title
    This is the danger admonition body

!!! success ""
    This is the success admonition body, with stripped out title and icon

    As shown below, there are plenty of other types to choose from !
```

A detailed description of the HTML code generated by the Markdown snippet above is provided in the last sections, starting from [Customization](#Customizing Admonitions).

Default Types Of Admonitions

Below is a list of the *default admonitions* types as given in the extension's [official documentation](#). Note that each icon is assigned to one and only one color. The converse is not true, though, as

each color is assigned to multiple icons. This behavior may be changed by creating your own custom types, as explained in the next section.

Note

Tldr | Abstract | Summary

Info | Todo

Tip | Hint

Done | Check | Success

Faq | Help | Question

Warning

Attention

Caution

Fail | Missing | Failure

Danger

Error

Bug

Snippet | Example

Cite | Quote

Customizing Admonitions

Admonition can be customized using CSS. To understand how, we first look at the HTML structure of an admonition. Then, we show how to implement a custom *type* and how to customize its look using CSS.

Transpiled HTML

The HTML structure of an admonition consists of a *division* container in which the header and every “chunk” of text in the body are considered as two separate *paragraphs*. Only the *division* and the title *paragraph*, are assigned a CSS class; the body *paragraph* is not. For example, the following markdown code :

```
!!! note The Note Title is considered as a paragraph
    And so is also the note's body, with the difference that is not assigned a CSS class

    It is worth noting that the body can be made of several paragraphs, as shown below
```

will generate *note* admonition with both a title and a body. The corresponding HTML code will then consist three distinct `<p>` elements within a `<div>` . More precisely, the generated HTML code will be:

```
<div class="admonition note">
<p class="admonition-title">The Note Title is considered as a paragraph</p>
<p>And so is also the note's body, with the difference that is not assigned a CSS class</p>
<p>It is worth noting that the body can be made of several paragraphs, as shown below</p>
</div>
```

Custom Types Of Admonitions

According to the previous section, getting your own custom types is then as simple as invoking your own *type* and defining the corresponding CSS class according to the desired look. For example, creating the “foo” *type* simply amounts to using the following markdown code:

```
```markdown
!!! foo The Foo Title This is the **foo** admonition body
```
```

and defining the corresponding CSS class to take control over the following HTML code:

```
<div class="admonition foo">
  <p class="admonition-title">The Foo Title</p>
  <p>This is the <strong>foo</strong> admonition body</p>
</div>
```

The following CSS code snippet shows how to customize the look of the “foo” admonition:

```
.admonition.foo {
  border: 1px solid #000000;
  border-radius: 0.5em;
  padding: 0.5em;
  margin: 1em 0;
  background-color: #f0f0f0;
  color: #000000;
}
.admonition.foo .admonition-title {
  font-weight: bold;
  text-transform: uppercase;
  font-size: 90%;
  margin: 0;
  padding: 0;
  border: 0;
  background-color: transparent;
  color: #000000;
}
```

Customizing Admonitions’ Styles

Last but not least, a custom CSS must be defined to take control over the HTML code generated by the extension. Due to the fact that only the `div container` contains and the `title paragraph` are assigned a CSS class, the CSS code to use is not so trivial. Indeed:

- `Div containers` are easy to select as they are simply any `div` elements with the `admonition` class.
- `Title paragraph` are the `p` elements with the `admonition-title` class **that are within the above div container**.
- `Body paragraphs` are the `p` elements that are within the above `div container` but that **do not have the `admonition-title` class**. That last condition is not easy to implement using CSS, since there are no such selectors as “not” or “except”. The easiest way to do it is to make sure that `body paragraphs` are styled **above** the `title paragraph` and that the latter override every style that is not desired.

Here is an example of a working CSS code snippet that implements the above rules:

```
.admonition.foo {
  border: 1px solid gray;
  border-radius: 0.5em;
  padding: 0.5em;
  margin: 1em 0;
  color: #323232;
}
div.admonition.foo p {
  margin: 0;
  padding: 0;
  border: 0;
  background-color: transparent;
  color: #lightyellow;
}
.admonition.foo .admonition-title {
  font-weight: bold;
  text-variant: small-caps;
  font-size: 90%;
  margin: 0;
  padding: 0;
  border: 0;
  background-color: transparent;
  color: #white;
}
```