

- **context** (`MarkdownContext`) – The markdown context.
- **text** (`str` / `Text`)

Return type

None

class `rich.markdown.UnknownElement`

An unknown element.

Hopefully there will be no unknown elements, and we will have a `MarkdownElement` for everything in the document.

23.14 rich.markup

class `rich.markup.Tag(name, parameters)`

A tag in console markup.

Parameters

- **name** (`str`)
- **parameters** (`str` / `None`)

property `markup: str`

Get the string representation of this tag.

name: `str`

The tag name. e.g. ‘bold’.

parameters: `str` | `None`

Any additional parameters after the name.

`rich.markup.escape(markup, _escape=<built-in method sub of re.Pattern object>)`

Escapes text so that it won’t be interpreted as markup.

Parameters

- **markup** (`str`) – Content to be inserted in to markup.
- **_escape** (`Callable[[Callable[[Match[str]], str], str], str]`)

Returns

Markup with square brackets escaped.

Return type`str``rich.markup.render(markup, style="", emoji=True, emoji_variant=None)`Render console markup in to a `Text` instance.**Parameters**

- **markup** (`str`) – A string containing console markup.
- **style** (`str` / `Style`) – (`Union[str, Style]`): The style to use.
- **emoji** (`bool`, *optional*) – Also render emoji code. Defaults to `True`.
- **emoji_variant** (`str`, *optional*) – Optional emoji variant, either “text” or “emoji”. Defaults to `None`.