

**Return type***Text***append\_text(*text*)**

Append another Text instance. This method is more performant than Text.append, but only works for Text.

**Parameters**

**text** (*Text*) – The Text instance to append to this instance.

**Returns**

Returns self for chaining.

**Return type***Text***append\_tokens(*tokens*)**

Append iterable of str and style. Style may be a Style instance or a str style definition.

**Parameters**

**tokens** (*Iterable[Tuple[str, Optional[StyleType]]]*) – An iterable of tuples containing str content and style.

**Returns**

Returns self for chaining.

**Return type***Text***apply\_meta(*meta*, *start*=0, *end*=None)**

Apply metadata to the text, or a portion of the text.

**Parameters**

- **meta** (*Dict[str, Any]*) – A dict of meta information.
- **start** (*int*) – Start offset (negative indexing is supported). Defaults to 0.
- **end** (*Optional[int]*, *optional*) – End offset (negative indexing is supported), or None for end of text. Defaults to None.

**Return type**

None

**classmethod assemble(\**parts*, *style*='', *justify*=None, *overflow*=None, *no\_wrap*=None, *end*='\n', *tab\_size*=8, *meta*=None)**

Construct a text instance by combining a sequence of strings with optional styles. The positional arguments should be either strings, or a tuple of string + style.

**Parameters**

- **style** (*Union[str, Style]*, *optional*) – Base style for text. Defaults to ''.
- **justify** (*str*, *optional*) – Justify method: “left”, “center”, “full”, “right”. Defaults to None.
- **overflow** (*str*, *optional*) – Overflow method: “crop”, “fold”, “ellipsis”. Defaults to None.
- **no\_wrap** (*bool*, *optional*) – Disable text wrapping, or None for default. Defaults to None.
- **end** (*str*, *optional*) – Character to end text with. Defaults to “\n”.