

- **text** (`str`) – A piece of text.
- **style** (`Style`, optional) – An optional style to apply to the text.
- **control** (`Tuple[ControlCode]`, `optional`) – Optional sequence of control codes.

cell_length

The cell length of this Segment.

Type

`int`

classmethod adjust_line_length(*line, length, style=None, pad=True*)

Adjust a line to a given width (cropping or padding as required).

Parameters

- **segments** (`Iterable[Segment]`) – A list of segments in a single line.
- **length** (`int`) – The desired width of the line.
- **style** (`Style`, `optional`) – The style of padding if used (space on the end). Defaults to None.
- **pad** (`bool`, `optional`) – Pad lines with spaces if they are shorter than *length*. Defaults to True.
- **line** (`List[Segment]`)

Returns

A line of segments with the desired length.

Return type

`List[Segment]`

classmethod align_bottom(*lines, width, height, style, new_lines=False*)

Aligns render to bottom (adds extra lines above as required).

Args:

lines (`List[List[Segment]]`): A list of lines. *width* (`int`): Desired width. *height* (`int, optional`): Desired height or None for no change. *style* (`Style`): Style of any padding added. Defaults to None. *new_lines* (`bool, optional`): Padded lines should include ““.

“. Defaults to False.

Returns:

`List[List[Segment]]`: New list of lines.

Parameters

- **lines** (`List[List[Segment]]`)
- **width** (`int`)
- **height** (`int`)
- **style** (`Style`)
- **new_lines** (`bool`)

Return type

`List[List[Segment]]`