

## 2.6 The row function

The `row` function can be used to transform or style a whole row at a time. For example, `row([a & b & c], map: strong)` styles the row using **strong**. The row function is entirely optional. It has three arguments:

- `map` - apply a function to each element
- `imap` - apply a function to each element, and also receive its index
- `cell` - set a dictionary of cell properties, e.g. `fill` and `stroke` on the whole row.

It is recommended to use the style `row([...], map: ..)` with the row markup as the first argument so that rows align well to each other.

**Example:** Use `map` or `imap` to transform or style the cells of a row.

*Document Result*

0	1	2	3	4	5
$\alpha^0$	$\beta^1$	$\gamma^2$	$\delta^3$	$\epsilon^4$	$\zeta^5$
a	b	c	d	e	f

*Input*

```
#import "@preview/rowmantic:0.3.0": rowtable, row
/// Draw cell with a color between `from` and `to`
#let colorcells(index, elt, from: yellow, to: red, step: 20%) = {
  let fr = index * step
  let fill = color.mix((to, fr), (from, 100% - fr)).lighten(50%)
  table.cell(fill: fill, elt)
}
#rowtable(
  separator: ", ", align: center, inset: 0.7em,
  row([0, 1, 2, 3, 4, 5 ], map: strong),
  row($alpha, beta, gamma, delta, epsilon, zeta$, imap: (idx, elt) => $elt^idx$),
  row([a, b, c, d, e, f ], imap: colorcells),
)
```

**Example:** Use `map` to highlight empty entries, and `cell` to set background color.

*Document Result*

A		C		E	F
G	H		J	K	

*Input*

```
/// Draw cell with thick stroke if not empty, and red background if empty
#let markempty(elt) = {
  if elt != [] and elt != none {
    table.cell(stroke: 2pt, elt)
  } else {
    table.cell(fill: red.lighten(80%), elt)
  }
}
#rowtable(
  separator: "& ", align: center, inset: 0.7em,
  row([ A & & C & & E & F ], map: markempty),
  row([ G & H & & J & K & ], map: markempty),
  row([ ~ & ], cell: (fill: gray)),
)
```