

SAPIANS DESIGN SYSTEM

Reference Manual

API reference generated from source with tidy · v1.0.0

Contents

1 Components	3
1.1 Header	3
1.2 Code Box	5
1.3 Callout Cards	6
1.4 Split Card	8
2 Dynamic slides	10
2.1 Overlays & Steps	10
2.2 Speaker Notes (pdfpc)	14
3 Slides	16
3.1 Slide Deck	16
4 Layouts	34
4.1 Scientific Article	34
4.2 Technical Report	35

1 Components

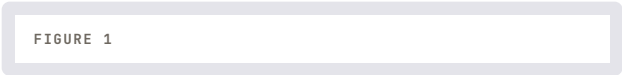
The building blocks used inside slides, papers, and reports: headers and labels, code boxes, callout cards, and the numbered/definition/comparison row helpers. Every example below is compiled live against the actual package, so the rendered preview always matches the current source.

1.1 Header

1.1.1 caps-label

An uppercase, wide-tracked label in the muted secondary tone. Use it for figure/table captions, axis labels, or any small annotation that should read as metadata rather than body text.

```
#box(fill: white, inset: 2pt)[#caps-label("figure 1")]
```



1.1.1.1 Parameters

```
caps-label(  
  text-content: str,  
  dark: bool  
) -> content
```

1.1.1.1.1 text-content str

Text to render, upper-cased automatically.

1.1.1.1.2 dark bool

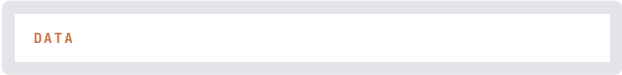
Whether the label sits on a dark canvas; picks the light-on-dark muted tone instead of the dark-on-light one.

Default: `false`

1.1.2 kicker

A tiny uppercase tag in the SAPIANS terracotta accent color, tracked wide. Use it to label a slide region ("DATA", "METHOD", ...) the same way a kicker introduces a headline.

```
#box(fill: white, inset: 2pt)[#kicker("Data")]
```



1.1.2.1 Parameters

```
kicker(  
  text-content: str,  
  dark: bool  
) -> content
```

1.1.2.1.1 text-content str

Text to render, upper-cased automatically.

1.1.2.1.2 dark bool

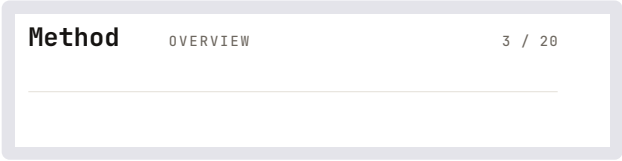
Accepted for symmetry with the other header helpers; the kicker is always rendered in terracotta and this flag currently has no visual effect.

Default: `false`

1.1.3 slide-header

Draws the fixed header strip at the top of a slide: a bold title, an optional uppercase section tag, an optional right-aligned page counter, and a hairline rule underneath. Call it once per slide, right after the canvas background is set, so title/section/counter placement stays identical across the whole deck.

```
#box(width: 70mm, height: 14mm, fill: white)[
  #slide-header(title: "Method", section: "Overview",
    counter: "3 / 20")
]
```



1.1.3.1 Parameters

```
slide-header(
  title: str,
  section: str,
  counter: str,
  dark: bool
) -> content
```

1.1.3.1.1 title str

Slide title, shown bold at the top-left.

Default: ""

1.1.3.1.2 section str

Uppercase section tag shown next to the title, in a muted tone. Omitted entirely when empty.

Default: ""

1.1.3.1.3 counter str

Page counter text shown at the top-right, e.g. "3 / 20". Omitted entirely when empty.

Default: ""

1.1.3.1.4 dark bool

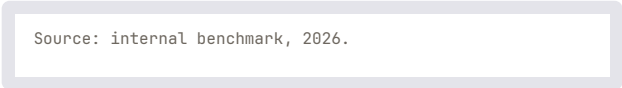
Whether the header sits on a dark canvas; swaps the title, section, and hairline colors for the dark palette.

Default: false

1.1.4 small-text

Small secondary body text in the muted tone, for captions, footnotes, or source attributions that should stay visually quiet next to the primary content.

```
#box(fill: white, inset: 2pt)[
  #small-text("Source: internal benchmark, 2026.")
]
```



1.1.4.1 Parameters

```
small-text(
  text-content: str content,
  dark: bool
) -> content
```

1.1.4.1.1 text-content str or content

Content to render at the small secondary size.

1.1.4.1.2 dark `bool`

Whether the text sits on a dark canvas; picks the light-on-dark muted tone instead of the dark-on-light one.

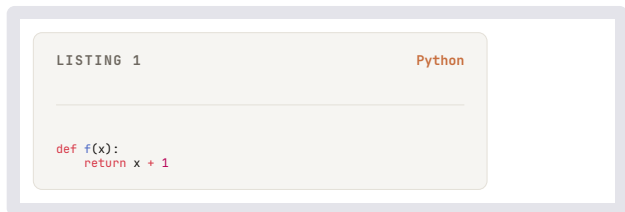
Default: `false`

1.2 Code Box

1.2.1 code-box

A bordered, monospaced code container with an optional title bar showing a label and a language tag. Use it whenever a slide, paper, or report needs to show source code as a self-contained block rather than inline `raw` text.

```
#code-box(width: 60mm, title: "Listing 1", lang:
"Python")
  #raw("def f(x):\n    return x + 1", lang: "python",
block: true)
]
```



1.2.1.1 Parameters

```
code-box(
  body: content,
  width: auto relative,
  height: auto relative,
  title: none str,
  lang: none str,
  dark: bool
) -> content
```

1.2.1.1.1 body `content`

Code (or any content) shown inside the box.

1.2.1.1.2 width `auto` or `relative`

Box width, forwarded to the underlying `block`.

Default: `100%`

1.2.1.1.3 height `auto` or `relative`

Box height, forwarded to the underlying `block`.

Default: `auto`

1.2.1.1.4 title `none` or `str`

Uppercase label shown in the title bar, e.g. "Listing 1". The title bar (and its hairline rule) is omitted entirely when `none`.

Default: `none`

1.2.1.1.5 lang `none` or `str`

Language tag shown at the right of the title bar in terracotta, e.g. "Python". Only shown when `title` is also set.

Default: `none`

1.2.1.1.6 dark `bool`

Whether the box sits on a dark canvas; swaps background, border, and text colors for the dark palette.

Default: `false`

1.3 Callout Cards

1.3.1 accent-card

A flush-left card with a terracotta accent rule down its left edge and an optional bold title. Lighter-weight than `dark-card()` and `light-card()` — use it for an inline note, caveat, or hypothesis statement that should stand out without a full card treatment.

```
#accent-card(width: 60mm, title: "Hypothesis")[
  Local surrogates approximate  $f$  only near  $x$ .
]
```

Hypothesis

Local surrogates approximate f only near x .

1.3.1.1 Parameters

```
accent-card(
  body: content,
  title: none str,
  width: auto relative
) -> content
```

1.3.1.1.1 body `content`

Card content.

1.3.1.1.2 title `none` or `str`

Bold title shown above the body, in terracotta. Omitted entirely when `none`.

Default: `none`

1.3.1.1.3 width `auto` or `relative`

Card width, forwarded to the underlying `block`.

Default: `100%`

1.3.2 dark-card

A high-contrast card on the SAPIANS dark fill, with an optional terracotta kicker above the body. Use it to call out a key insight or result that should visually “pop” against a light slide or page.

```
#dark-card(width: 60mm, kicker-title: "Insight")[
  Local surrogates approximate  $f$  only near  $x$ .
]
```

INSIGHT

Local surrogates approximate f only near x .

1.3.2.1 Parameters

```
dark-card(
  body: content,
  kicker-title: none str,
  width: auto relative,
  height: auto relative,
  inset: relative dictionary
) -> content
```

1.3.2.1.1 body content

Card content.

1.3.2.1.2 kicker-title none or str

Uppercase terracotta kicker shown above the body. Omitted entirely when `none`.

Default: `none`

1.3.2.1.3 width auto or relative

Card width, forwarded to the underlying `block`.

Default: `100%`

1.3.2.1.4 height auto or relative

Card height, forwarded to the underlying `block`.

Default: `auto`

1.3.2.1.5 inset relative or dictionary

Inner padding, forwarded to the underlying `block`.

Default: `3.2mm`

1.3.3 light-card

A subtly elevated card on the light card background, with a hairline border and an optional terracotta kicker. This is the light-canvas counterpart of `dark-card()` — use it for supporting content that should stay visually quiet on a light slide or page.

```
#light-card(width: 60mm, kicker-title: "Insight")[
  Local surrogates approximate $f$ only near $x$.
]
```

INSIGHT

Local surrogates approximate f only near x .

1.3.3.1 Parameters

```
light-card(
  body: content,
  kicker-title: none str,
  width: auto relative,
  height: auto relative,
  inset: relative dictionary,
  fill: color,
  stroke: stroke
) -> content
```

1.3.3.1.1 body `content`

Card content.

1.3.3.1.2 kicker-title `none` or `str`

Uppercase terracotta kicker shown above the body. Omitted entirely when `none`.

Default: `none`

1.3.3.1.3 width `auto` or `relative`

Card width, forwarded to the underlying `block`.

Default: `100%`

1.3.3.1.4 height `auto` or `relative`

Card height, forwarded to the underlying `block`.

Default: `auto`

1.3.3.1.5 inset `relative` or `dictionary`

Inner padding, forwarded to the underlying `block`.

Default: `3.2mm`

1.3.3.1.6 fill `color`

Card background fill.

Default: `sapians-card-bg`

1.3.3.1.7 stroke `stroke`

Card border stroke.

Default: `stroke-light`

1.4 Split Card

1.4.1 contrast-pair

A fixed-height two-up comparison: a muted “not this” card next to a high-contrast dark “this” card, each with its own uppercase label. Use it to contrast a naive approach against the recommended one.

```
#box(width: 90mm)[
  #contrast-pair([Fits the whole dataset], [Fits near
the query point])
]
```



1.4.1.1 Parameters

```
contrast-pair(
  not-this-content: str content,
  this-content: str content,
```



```

not-this-label: str,
this-label: str,
this-sub: none str
) -> content

```

1.4.1.1.1 not-this-content `str` or `content`

Content for the muted “not this” card.

1.4.1.1.2 this-content `str` or `content`

Content for the dark, high-contrast “this” card.

1.4.1.1.3 not-this-label `str`

Uppercase label on the “not this” card.

Default: “NOT THIS”

1.4.1.1.4 this-label `str`

Uppercase label on the “this” card.

Default: “THIS”

1.4.1.1.5 this-sub `none` or `str`

Uppercase sub-label shown below the “this” content. Omitted entirely when `none`.

Default: “LOCAL ≠ GLOBAL”

1.4.2 def-row

A single-line definition row: an uppercase terracotta kicker on the left and a bold value on the right, both inside a bordered card. Use it for compact key/value pairs, e.g. labeling a formula or a named quantity.

```
#def-row("Black box", "f(x)", width: 60mm)
```

BLACK BOX

f(x)

1.4.2.1 Parameters

```

def-row(
  kicker-text: str,
  value-text: str content,
  width: auto relative
) -> content

```

1.4.2.1.1 kicker-text `str`

Uppercase kicker on the left.

1.4.2.1.2 value-text `str` or `content`

Bold value on the right.

1.4.2.1.3 width `auto` or `relative`

Row width, forwarded to the underlying block.

Default: `100%`

1.4.3 step-item

A numbered key item: a two-digit terracotta index next to a title and an optional description. Use it for ordered lists of steps, findings, or takeaways where the sequence itself carries meaning.

```
#step-item(
  1,
  "Approximate the model",
  description: "Fit a local surrogate around the
point of interest.",
)
```

**01 Approximate
the
model**

Fit
a
local
surrogate
around
the
point
of
interest.

1.4.3.1 Parameters

```
step-item(
  number: int str,
  title: str content,
  description: none str content,
  dark: bool
) -> content
```

1.4.3.1.1 number `int` or `str`

Index shown as a zero-padded two-digit number when given as an `int` (e.g. 1 becomes "01"); any other value is shown verbatim.

1.4.3.1.2 title `str` or `content`

Item title, shown bold whenever description is set.

1.4.3.1.3 description `none` or `str` or `content`

Supporting text shown below the title, in the muted tone. Omitted entirely when `none`.

Default: `none`

1.4.3.1.4 dark `bool`

Whether the item sits on a dark canvas; swaps title and description colors for the dark palette.

Default: `false`

2 Dynamic slides

Incremental reveal (`pause`, `uncover`, `only`), handout mode, and the pdfpc speaker-notes toolchain. Portions adapted from touying (MIT) and inspired by minideck/polylux — see the module headers for attribution.

2.1 Overlays & Steps

2.1.1 check-visible

Check whether a subslide index is visible under a visibility specification. `check-visible(3, "2-")` is `true`.

2.1.1.1 Parameters

```
check-visible(
  idx: int,
  spec: int array str dictionary
) -> bool
```

2.1.1.1.1 idx int

The subslide being rendered, 1-based.

2.1.1.1.2 spec int or array or str or dictionary

Which subslides the content belongs to. A single integer selects one subslide, an array selects several, a string spells out ranges ("-2, 4, 6-8, 10-") and may start with ! to negate them, and a dictionary bounds the range with beginning and/or until.

2.1.2 is-pause

Whether `item` is the pause sentinel.

2.1.2.1 Parameters

```
is-pause(item: any) -> bool
```

2.1.2.1.1 item any

The array entry to test.

2.1.3 last-required-subslide

The highest subslide index a specification requires, i.e. how many steps a slide containing it must have.

2.1.3.1 Parameters

```
last-required-subslide(spec: int array str dictionary) -> int
```

2.1.3.1.1 spec int or array or str or dictionary

The specification, in any of the forms `check-visible` accepts.

2.1.4 only

Like `uncover`, but the body is removed from the layout on the subslides where it is not visible.

2.1.4.1 Parameters

```
only(
  spec: int array str dictionary,
  body: content
) -> content
```

2.1.4.1.1 spec int or array or str or dictionary

Which subslides show the body.

2.1.4.1.2 body content

The content to show.

2.1.5 pause-levels

Strip the pause sentinels out of an array and report, for every surviving item, the subslide it appears on.

Returns (items: array, levels: array, steps: int) where levels.at(i) is the 1-based subslide index from which items.at(i) is visible, and steps is the number of subslides the array requires.

2.1.5.1 Parameters

`pause-levels`(arr: array) -> dictionary

2.1.5.1.1 arr array

The slot array, with or without pause sentinels.

2.1.6 resolve-handout

Resolve the handout flag. `--input handout=true` on the command line turns handout mode on whatever the document says; `auto` leaves the decision entirely to the command line.

2.1.6.1 Parameters

`resolve-handout`(handout: bool auto) -> bool

2.1.6.1.1 handout bool or auto

The engine's handout argument.

2.1.7 reveal-at

Show body from subslide level on, keeping its layout box on the earlier subslides. Level 1 short-circuits to plain content, so static slides never pay for a context.

2.1.7.1 Parameters

```
reveal-at(
  level: int,
  body: content,
  hider: function
) -> content
```

2.1.7.1.1 level int

First subslide on which the body is visible, 1-based.

2.1.7.1.2 body content

The content to reveal.

2.1.7.1.3 hider function

How to hide the body before level; hide keeps its box, pass `_` $\Rightarrow$ none to drop it from the layout.

Default: hide

2.1.8 split-at-pauses

Split an array at every `pause` sentinel into cumulative groups: the first group holds the items before the first pause, the second one those items plus the items up to the second pause, and so on. The last group is the full array without the sentinels.

Example: `split-at-pauses((a, b, pause, c))` is `((a, b), (a, b, c))`.

2.1.8.1 Parameters

```
split-at-pauses(arr: array) -> array
```

2.1.8.1.1 arr `array`

The slot array, with or without `pause` sentinels.

2.1.9 stepped-slide

Render a slide as a sequence of subslides.

`render` is called with the 1-based subslide index and must return the complete slide content (i.e. it calls `slide()`, which emits the page).

With `steps: auto` the slide is rendered once, and the number of steps that first rendering asked for (through `uncover`, `only` or `update-steps`) decides how often it is repeated — the `minideck _slide-auto` trick. Pass an integer to `steps` to pin the count.

In handout mode a single page is emitted with every step revealed, which is exactly the last subslide for pause-driven content.

2.1.9.1 Parameters

```
stepped-slide(
  steps: auto int,
  render: function
) -> content
```

2.1.9.1.1 steps `auto` or `int`

Number of subslides, or `auto` to take the number the first rendering asked for.

Default: `auto`

2.1.9.1.2 render `function`

Called with the 1-based subslide index; must return the whole slide, page included.

2.1.10 uncover

Uncover `body` on the subslides matched by `spec`, keeping its layout box on the others. The step count of the enclosing slide grows automatically, so `#uncover("3-") [..]` inside a slide body turns it into a three-step slide.

2.1.10.1 Parameters

```
uncover(
  spec: int array str dictionary,
  body: content,
  hider: function
) -> content
```

2.1.10.1.1 spec `int` or `array` or `str` or `dictionary`

Which subslides show the body, in any of the forms `check-visible` accepts.

2.1.10.1.2 body `content`

The content to uncover.

2.1.10.1.3 hider `function`

How to hide the body elsewhere.

Default: `hide`

2.1.11 update-by-pause

Pure state update: advance the pause cursor by one and make sure the slide has at least one more step than the new cursor value. The building block behind pause-driven steps, kept public for custom families.

2.1.11.1 Parameters

`update-by-pause()` -> `content`

2.1.12 update-steps

Pure state update: make sure the current slide has at least `n` steps. Emit it anywhere inside a slide — it renders nothing and needs no context, which is what keeps layout convergence stable.

2.1.12.1 Parameters

`update-steps(n: int)` -> `content`

2.1.12.1.1 n `int`

The number of steps the slide must have at least.

2.1.13 logical-slide

Logical slide number: one per slide, no matter how many subslides it is rendered as. `counter(page)` is deliberately left untouched, so the two diverge exactly the way pdfpc expects.

2.1.14 pause

Sentinel value placed between the items of a slot array (`definitions:`, `steps:`, `points:`, `items:`) to split the slide into steps.

`#slide-definition(definitions: (a, b, pause, c))`

2.2 Speaker Notes (pdfpc)**2.2.1 pdfpc-config**

Document-wide pdfpc configuration. Call it before the first slide.

`#pdfpc-config(duration-minutes: 30, last-minutes: 5)`

2.2.1.1 Parameters

```
pdfpc-config(
  duration-minutes: none int,
  start-time: none str datetime,
  end-time: none str datetime,
  last-minutes: none int,
  note-font-size: none int float,
```

```
    disable-markdown: bool
) -> content
```

2.2.1.1.1 duration-minutes `none` or `int`

Length of the talk in minutes.

Default: `none`

2.2.1.1.2 start-time `none` or `str` or `datetime`

Wall-clock start time, HH:MM when given as a string.

Default: `none`

2.2.1.1.3 end-time `none` or `str` or `datetime`

Wall-clock end time, HH:MM when given as a string.

Default: `none`

2.2.1.1.4 last-minutes `none` or `int`

How many minutes before the end the timer turns red.

Default: `none`

2.2.1.1.5 note-font-size `none` or `int` or `float`

Font size of pdfpc's notes pane.

Default: `none`

2.2.1.1.6 disable-markdown `bool`

Render the notes as plain text instead of markdown.

Default: `false`

2.2.2 pdfpc-file

Collect every `<pdfpc>` marker into the `pdfpcFormat: 2` document and publish it as `<pdfpc-file>`. Installed once by the slide engine.

2.2.2.1 Parameters

```
pdfpc-file(loc: location) -> content
```

2.2.2.1.1 loc `location`

The location the collector runs at, i.e. `here()`.

2.2.3 pdfpc-page-markers

The per-page markers. Emitted by `slide()` inside the page, so that `here().page()` is the physical page the markers belong to.

2.2.3.1 Parameters

```
pdfpc-page-markers(note: none str raw) -> content
```

2.2.3.1.1 note none or str or raw

Speaker note for this page.

Default: `none`

2.2.4 speaker-note

Emit a speaker note for the page it appears on. Every slide family also takes a `note:` argument, which calls this for you.

2.2.4.1 Parameters

```
speaker-note(note: str raw) -> content
```

2.2.4.1.1 note str or raw

The note. A raw block is unwrapped and trimmed, so a fenced markdown block can carry the note verbatim.

2.2.5 end-slide

Mark the current slide as the last one of the presentation proper; pdfpc treats the slides after it as backup material.

2.2.6 save-slide

Mark the current slide as pdfpc's "saved" slide.

2.2.7 hidden-slide

Hide the current slide in pdfpc's overview.

3 Slides

The slide deck layout: the `sapians-slides` engine and `slide` escape hatch, followed by the 10 fixed slide families a SAPIANS deck is composed from — each one enforcing a single dominant idea per slide, so a deck reads as a sequence of ideas rather than a wall of layouts. See `docs/slides.md` for the full taxonomy, the composition rules (open dark, close dark; one `slide-contrast` per talk at most; ...), and worked examples. Every function here sets page-level rules internally (a full `page()`), so — like the layouts below — none of them can be called from inside another container or shown as a live doc example; each description carries a static usage snippet instead, and the live, rendered preview lives in the visual-regression gallery (`packages/typst/tests/slides/`) and the slides template (`templates/slides-deck/template/main.typ`).

3.1 Slide Deck

3.1.1 sapians-slides

The SAPIANS slide engine: sets the deck-wide page geometry, base typography, and the dynamics/pdfpc state, then renders `body`. Apply it once per document with a `show` rule, before any `slide/slide-*` call — it sets page-level rules, so it cannot be called from inside another container (a card, a box, ...) or from a live doc example; see the usage snippet below.

```
#import "@preview/sapians-slides:0.2.0": *
```



```
#show: sapians-slides.with(
  title: "Deck Title",
  author: "Author Name",
  aspect-ratio: "16-9",
)
```

```
#slide-cover(title: "SAPIANS", subtitle: "A one-line thesis.")
```

3.1.1.1 Parameters

```
sapians-slides(
  title: str,
  author: none or str,
  lang: str,
  aspect-ratio: str,
  handout: bool,
  body: content
) -> content
```

3.1.1.1.1 title str

Document title, forwarded to `set document(title: ...)`.

Default: "SAPIANS Presentation"

3.1.1.1.2 author none or str

Document author, forwarded to `set document(author: ...)`. Defaults to "SAPIANS" when none.

Default: none

3.1.1.1.3 lang str

Document language, forwarded to `set text(lang: ...)`.

Default: "en"

3.1.1.1.4 aspect-ratio str

Page aspect ratio: "16-9" (160×90mm) or "16-10" (160×100mm). In 16:10 the slide content keeps its 16:9 box, centred vertically in a 90mm band, so no family has to know the page size.

Default: "16-9"

3.1.1.1.5 handout bool

Whether to render the deck in handout mode: every slide collapses to a single page with all of its steps revealed. Can also be turned on from the command line with `--input handout=true`; the two are ORed together, so the command line can force handout mode on but never off.

Default: false

3.1.1.1.6 body content

Deck content: the sequence of `slide/slide-*` calls. Supplied automatically by `#show: sapians-slides.with(...)`.

3.1.2 slide

The raw slide canvas with the theme applied but none of the 10 family layouts. Use it for the rare slide the family taxonomy doesn't cover — see `docs/slides.md` for the taxonomy — and consider proposing a new family (see

the design-change issue form) if you reach for it twice. Like the families, it sets page-level rules internally, so it cannot be called from inside another container (a card, a box, ...) or from a live doc example; see the usage snippet below.

```
#slide(dark: true, note: "Close on the one sentence to remember.")[
  #slide-header(title: "TOPIC", counter: "20 / 20", dark: true)
  #place(top + left, dx: 2mm, dy: 20mm)[
    #text(size: 16pt, weight: "bold", fill: sapians-text-light)[
      A closing line the taxonomy has no slot for.
    ]
  ]
]
```

3.1.2.1 Parameters

```
slide(
  dark: bool,
  note: none or str,
  subslides: auto or int,
  body: content
) -> content
```

3.1.2.1.1 dark bool

Whether the slide sits on the dark canvas; swaps the background and text colors for the dark palette.

Default: `false`

3.1.2.1.2 note none or str

Speaker note exported to the pdfpc sidecar as this slide's note. Omitted entirely when `none`.

Default: `none`

3.1.2.1.3 subslides auto or int

Number of subslides (steps) to render. With `auto`, the `uncover()` and `only()` calls (or a manual call to `update-steps()`) inside `body` decide the count; pass an integer to pin it explicitly.

Default: `auto`

3.1.2.1.4 body content

Slide content, placed directly on the page.

3.1.3 slide-contrast

Family 08: contrast slide (Not This vs This) — a headline followed by a muted “not this” card next to a high-contrast dark “this” card. Use it to correct a common misreading; per the composition rule, aim for at most one per talk — it’s a scalpel, not a format. It sets page-level rules internally, so it cannot be called from inside another container or from a live doc example; see the usage snippet below.

```
#slide-contrast(
  title: "GUIDELINES",
  section: "DECISION",
  counter: "05 / 06",
  hero: "What do we optimize for?",
  not-this-content: [A common but wrong interpretation.],
  this-content: [The correct interpretation, stated as concretely.],
  not-this-label: "NOT THIS",
  this-label: "THIS",
  this-sub: "CLARITY > NOISE",
)
```

3.1.3.1 Parameters

```
slide-contrast(
  title: str,
  section: str,
  counter: str,
  hero: str,
  not-this-content: str content,
  this-content: str content,
  not-this-label: str,
  this-label: str,
  this-sub: none str,
  note: none str,
  subslides: auto int
) -> content
```

3.1.3.1.1 title str

Slide title, shown bold at the top-left by `slide-header`.

Default: ""

3.1.3.1.2 section str

Uppercase section tag shown next to the title.

Default: "INTERPRETATION"

3.1.3.1.3 counter str

Page counter text shown at the top-right, e.g. "08 / 10". Omitted entirely when empty.

Default: ""

3.1.3.1.4 hero str

Headline shown above the two cards. "\n" starts a new line.

Default: ""

3.1.3.1.5 not-this-content str or content

Content of the muted "not this" card.

Default: ""

3.1.3.1.6 this-content str or content

Content of the dark, high-contrast "this" card.

Default: ""

3.1.3.1.7 not-this-label str

Uppercase label on the "not this" card.

Default: "NOT THIS"

3.1.3.1.8 this-label str

Uppercase label on the "this" card.

Default: `"THIS"`

3.1.3.1.9 this-sub `none` or `str`

Uppercase sub-label shown below the "this" content. Omitted entirely when `none`.

Default: `"LOCAL ≠ GLOBAL"`

3.1.3.1.10 note `none` or `str`

Speaker note exported to the pdfpc sidecar for this slide. Omitted entirely when `none`.

Default: `none`

3.1.3.1.11 subslides `auto` or `int`

Number of subslides (steps) to render; with `auto` this family always resolves to a single step, since it has no reveal slot.

Default: `auto`

3.1.4 slide-cover

Family 01: cover slide — full-bleed title, subtitle, byline. Use it for the opening slide and for section breaks; per the deck-composition rule, open dark and close dark (see `slide-takeaway`), with every other family on the light canvas in between. It sets page-level rules internally, so it cannot be called from inside another container or from a live doc example; see the usage snippet below.

```
#slide-cover(
  title: "SAPIANS",
  subtitle: "A Clear, One-Line Statement\nof What This Deck Argues",
  author: "Author Name",
  affiliation: "Your Team or Institution",
  dark: true,
  note: "Welcome the audience and state the one thing to remember.",
)
```

3.1.4.1 Parameters

```
slide-cover(
  title: str,
  subtitle: str,
  author: str,
  affiliation: str,
  dark: bool,
  note: none str,
  subslides: auto int
) -> content
```

3.1.4.1.1 title `str`

Deck or slide title, shown large at the top-left.

Default: `""`

3.1.4.1.2 subtitle `str`

Supporting line shown under the title. Omitted entirely when empty; `"\n"` starts a new line.

Default: `""`

3.1.4.1.3 author `str`

Author name shown at the bottom-left. Omitted entirely when empty.

Default: `""`

3.1.4.1.4 affiliation `str`

Affiliation shown under the author, in the muted tone. Omitted entirely when empty.

Default: `""`

3.1.4.1.5 dark `bool`

Whether the slide sits on the dark canvas; swaps title, subtitle, author, and affiliation colors for the dark palette.

Default: `true`

3.1.4.1.6 note `none` or `str`

Speaker note exported to the pdfpc sidecar for this slide. Omitted entirely when `none`.

Default: `none`

3.1.4.1.7 subslides `auto` or `int`

Number of subslides (steps) to render; with `auto` this family always resolves to a single step, since it has no reveal slot.

Default: `auto`

3.1.5 slide-definition

Family 03: definition split slide — a headline concept on the left and a stack of (`kicker`, `value`) rows on the right. Use it for a concept plus its named parts (a term and its components, an acronym and its letters, ...). It sets page-level rules internally, so it cannot be called from inside another container or from a live doc example; see the usage snippet below.

```
#slide-definition(
  title: "TOPIC",
  section: "ARCHITECTURE",
  counter: "03 / 06",
  hero: "Name the core idea\nin one line.",
  explanation: "One sentence expanding the idea.",
  definitions: (
    ("PART ONE", "What it does"),
    ("PART TWO", "What it does"),
    pause,
    ("PART THREE", "What it does"),
    ("PART FOUR", "What it does"),
  ),
  note: "Walk through the first two, pause, then reveal the rest.",
)
```

3.1.5.1 Parameters

```
slide-definition(
  title: str,
  section: str,
  counter: str,
  hero: str,
  explanation: str,
```

```

definitions: array,
note: none str,
subslides: auto int
) -> content

```

3.1.5.1.1 title str

Slide title, shown bold at the top-left by `slide-header`.

Default: ""

3.1.5.1.2 section str

Uppercase section tag shown next to the title.

Default: "DEFINITION"

3.1.5.1.3 counter str

Page counter text shown at the top-right, e.g. "03 / 10". Omitted entirely when empty.

Default: ""

3.1.5.1.4 hero str

Headline concept name, shown large on the left column. "\n" starts a new line.

Default: ""

3.1.5.1.5 explanation str

Supporting sentence shown under `hero`, in the muted tone.

Default: ""

3.1.5.1.6 definitions array

Definition rows shown on the right, each a (`kicker`, `value`) pair — an uppercase tag and its bold value, e.g. ("PART ONE", "What it does"). May contain the `pause sentinel`: every `pause` starts a new step, and the rows after it keep their box but stay hidden until then.

Default: ()

3.1.5.1.7 note none or str

Speaker note exported to the pdfpc sidecar for this slide. Omitted entirely when `none`.

Default: `none`

3.1.5.1.8 subslides auto or int

Number of subslides (steps) to render. With `auto`, the number of `pause sentinels` in `definitions` decides the count; pass an integer to pin it explicitly.

Default: `auto`

3.1.6 slide-equation

Family 04: equation journal slide — one boxed formula followed by a numbered list of annotations. Use it for the single formula that deserves the whole room, walked through term by term. It sets page-level rules internally, so it cannot be called from inside another container or from a live doc example; see the usage snippet below.

```
#slide-equation(
  title: "TOPIC",
  section: "THE OBJECTIVE",
  counter: "04 / 10",
  hero: "A fixed equation slide.",
  equation: $ xi(x) = limits("arg min")_(g in G) cal(L)(f, g, pi_x) + Omega(g) $,
  steps: (
    "First step annotation.",
    pause,
    "Second step annotation.",
  ),
)
```

3.1.6.1 Parameters

```
slide-equation(
  title: str,
  section: str,
  counter: str,
  hero: str,
  equation: none content,
  steps: array,
  note: none str,
  subslides: auto int
) -> content
```

3.1.6.1.1 title str

Slide title, shown bold at the top-left by `slide-header`.

Default: ""

3.1.6.1.2 section str

Uppercase section tag shown next to the title.

Default: "THE OBJECTIVE"

3.1.6.1.3 counter str

Page counter text shown at the top-right, e.g. "04 / 10". Omitted entirely when empty.

Default: ""

3.1.6.1.4 hero str

Headline shown above the equation box. "\n" starts a new line.

Default: ""

3.1.6.1.5 equation none or content

The formula, shown centered inside a bordered, shaded box, e.g. $y = f(x)$. Omitted entirely when `none`.

Default: `none`

3.1.6.1.6 steps array

Annotation lines shown below the equation as a numbered list. May contain the `pause` sentinel: every `pause` starts a new step, and the annotations after it keep their box but stay hidden until then.

Default: `()`

3.1.6.1.7 note `none` or `str`

Speaker note exported to the pdfpc sidecar for this slide. Omitted entirely when `none`.

Default: `none`

3.1.6.1.8 subslides `auto` or `int`

Number of subslides (steps) to render. With `auto`, the number of `pause` sentinels in `steps` decides the count; pass an integer to pin it explicitly.

Default: `auto`

3.1.7 slide-evidence

Family 06: evidence graph / full-bleed slide — a short hero and subtext column next to a large `graphic`. Use it when a chart or figure IS the argument; per the composition rule, give `graphic` 60%+ of the slide, and reach for `slide-problem` with a `visual` instead if the chart is merely illustrative. It sets page-level rules internally, so it cannot be called from inside another container or from a live doc example; see the usage snippet below.

```
#slide-evidence(
  title: "TOPIC",
  section: "MECHANISM",
  counter: "06 / 10",
  hero: "A fixed evidence hero.",
  subtext: "Fixed evidence subtext.",
  graphic: image("fig-mechanism.svg", width: 70mm),
)
```

3.1.7.1 Parameters

```
slide-evidence(
  title: str,
  section: str,
  counter: str,
  hero: str,
  subtext: str,
  graphic: none content,
  note: none str,
  subslides: auto int
) -> content
```

3.1.7.1.1 title `str`

Slide title, shown bold at the top-left by `slide-header`.

Default: `""`

3.1.7.1.2 section `str`

Uppercase section tag shown next to the title.

Default: `"MECHANISM"`

3.1.7.1.3 counter `str`

Page counter text shown at the top-right, e.g. "06 / 10". Omitted entirely when empty.

Default: ""

3.1.7.1.4 hero `str`

Headline shown above `subtext`. "\n" starts a new line.

Default: ""

3.1.7.1.5 subtext `str`

Supporting sentence shown under `hero`, in the muted tone.

Default: ""

3.1.7.1.6 graphic `none` or `content`

The chart, figure, or diagram that carries the argument, placed to the right of the hero/subtext column. Omitted entirely when `none`.

Default: `none`

3.1.7.1.7 note `none` or `str`

Speaker note exported to the pdfpc sidecar for this slide. Omitted entirely when `none`.

Default: `none`

3.1.7.1.8 subslides `auto` or `int`

Number of subslides (steps) to render; with `auto` this family always resolves to a single step, since it has no reveal slot.

Default: `auto`

3.1.8 slide-index

Family 10: component index / summary slide — a 2×2 grid of kicker + title (+ optional description) cards. Use it for an agenda, a roadmap, or an architecture overview: any set of 3-4 named parts shown together rather than walked one at a time. It sets page-level rules internally, so it cannot be called from inside another container or from a live doc example; see the usage snippet below.

```
#slide-index(
  title: "SAPIANS",
  section: "ROADMAP",
  counter: "10 / 10",
  items: (
    ("01", "First item", "Fixed description."),
    pause,
    ("02", "Second item", "Fixed description."),
    ("03", "Third item"),
    ("04", "Fourth item"),
  ),
)
```

3.1.8.1 Parameters

```
slide-index(
  title: str,
  section: str,
```

```

    counter: str,
    items: array,
    note: none str,
    subslides: auto int
) -> content

```

3.1.8.1.1 title str

Slide title, shown bold at the top-left by `slide-header`.

Default: "SAPIANS"

3.1.8.1.2 section str

Uppercase section tag shown next to the title.

Default: "ROADMAP"

3.1.8.1.3 counter str

Page counter text shown at the top-right, e.g. "10 / 10". Omitted entirely when empty.

Default: ""

3.1.8.1.4 items array

Cards shown in a 2×2 grid, each a (kicker, title) or (kicker, title, description) tuple, e.g. ("01", "First item", "What it does") — the description is optional per item. May contain the `pause` sentinel: every `pause` starts a new step, and the cards after it keep their box but stay hidden until then.

Default: ()

3.1.8.1.5 note none or str

Speaker note exported to the pdfpc sidecar for this slide. Omitted entirely when `none`.

Default: `none`

3.1.8.1.6 subslides auto or int

Number of subslides (steps) to render. With `auto`, the number of `pause` sentinels in `items` decides the count; pass an integer to pin it explicitly.

Default: `auto`

3.1.9 slide-limitation

Family 07: limitation compare slide — a numbered list of points next to two labeled visuals and a conclusion line. Use it to compare two regimes side by side (narrow vs wide, before vs after, ...) or to state a limitation with the evidence for it. It sets page-level rules internally, so it cannot be called from inside another container or from a live doc example; see the usage snippet below.

```

#slide-limitation(
  title: "TOPIC",
  section: "LIMITS",
  counter: "07 / 10",
  hero: "Fixed limits hero.",
  points: ("First limit.", pause, "Second limit."),
  visual1: rect(width: 38mm, height: 26mm, fill: sapians-card-bg),

```

```

label1: "NARROW",
sub1: "kernel = 0.20",
visual2: rect(width: 38mm, height: 26mm, fill: sapians-card-bg),
label2: "WIDE",
sub2: "kernel = 0.60",
conclusion-title: "Fixed conclusion.",
conclusion-sub: "Fixed conclusion subtext.",
)

```

3.1.9.1 Parameters

```

slide-limitation(
  title: str,
  section: str,
  counter: str,
  hero: str,
  points: array,
  visual1: none content,
  label1: str,
  sub1: str,
  visual2: none content,
  label2: str,
  sub2: str,
  conclusion-title: str,
  conclusion-sub: str,
  note: none str,
  subslides: auto int
) -> content

```

3.1.9.1.1 title str

Slide title, shown bold at the top-left by `slide-header`.

Default: ""

3.1.9.1.2 section str

Uppercase section tag shown next to the title.

Default: "LIMITATIONS"

3.1.9.1.3 counter str

Page counter text shown at the top-right, e.g. "07 / 10". Omitted entirely when empty.

Default: ""

3.1.9.1.4 hero str

Headline shown above the point list. "\n" starts a new line.

Default: ""

3.1.9.1.5 points array

Bullet lines shown as a numbered list on the left. May contain the `pause` sentinel: every `pause` starts a new step, and the points after it keep their box but stay hidden until then.

Default: ()

3.1.9.1.6 visual1 none or content

First visual, shown above `label1/sub1`. Omitted entirely when `none`.

Default: `none`

3.1.9.1.7 `label1` `str`

Uppercase label under `visual1`.

Default: `"NARROW"`

3.1.9.1.8 `sub1` `str`

Small caption next to `label1`, in the muted tone.

Default: `"kernel = 0.20"`

3.1.9.1.9 `visual2` `none` or `content`

Second visual, shown above `label2/sub2`. Omitted entirely when `none`.

Default: `none`

3.1.9.1.10 `label2` `str`

Uppercase label under `visual2`.

Default: `"WIDE"`

3.1.9.1.11 `sub2` `str`

Small caption next to `label2`, in the muted tone.

Default: `"kernel = 0.60"`

3.1.9.1.12 `conclusion-title` `str`

Bold conclusion line shown under the two visuals.

Default: `""`

3.1.9.1.13 `conclusion-sub` `str`

Supporting sentence shown under `conclusion-title`, in the muted tone.

Default: `""`

3.1.9.1.14 `note` `none` or `str`

Speaker note exported to the pdfpc sidecar for this slide. Omitted entirely when `none`.

Default: `none`

3.1.9.1.15 `subslides` `auto` or `int`

Number of subslides (steps) to render. With `auto`, the number of `pause` sentinels in `points` decides the count; pass an integer to pin it explicitly.

Default: `auto`

3.1.10 slide-problem

Family 02: problem / hero statement slide — a headline claim, one supporting sentence, and a dark question card, with room for a supporting visual on the right. Use it to frame the problem and the question the deck answers, typically right after `slide-cover`. It sets page-level rules internally, so it cannot be called from inside another container or from a live doc example; see the usage snippet below.

```
#slide-problem(
  title: "TOPIC",
  section: "DIAGNOSIS",
  counter: "02 / 06",
  hero: "State the problem\nin one strong sentence.",
  subtext: "One supporting sentence explaining why it matters.",
  question-label: "THE QUESTION",
  question: "What is the single question\nthis deck answers?",
  visual: block(
    fill: sapians-white,
    stroke: stroke-light,
    radius: radius-sm,
    inset: 4mm,
    [A short, memorable statement that frames the argument.],
  ),
)
```

3.1.10.1 Parameters

```
slide-problem(
  title: str,
  section: str,
  counter: str,
  hero: str,
  subtext: str,
  question: str,
  question-label: str,
  visual: none content,
  note: none str,
  subslides: auto int
) -> content
```

3.1.10.1.1 title str

Slide title, shown bold at the top-left by `slide-header`.

Default: ""

3.1.10.1.2 section str

Uppercase section tag shown next to the title.

Default: "THE PROBLEM"

3.1.10.1.3 counter str

Page counter text shown at the top-right, e.g. "02 / 10". Omitted entirely when empty.

Default: ""

3.1.10.1.4 hero str

Headline claim, shown large under the header. "\n" starts a new line.

Default: ""

3.1.10.1.5 subtext str

Supporting sentence shown under `hero`, in the muted tone.

Default: ""

3.1.10.1.6 question str

The question shown inside the dark card.

Default: ""

3.1.10.1.7 question-label str

Uppercase kicker shown above `question` inside the dark card.

Default: "THE QUESTION"

3.1.10.1.8 visual none or content

Supporting visual placed to the right of the hero/question column (chart, card, diagram, ...). Omitted entirely when `none`.

Default: `none`

3.1.10.1.9 note none or str

Speaker note exported to the pdfpc sidecar for this slide. Omitted entirely when `none`.

Default: `none`

3.1.10.1.10 subslides auto or int

Number of subslides (steps) to render; with `auto` this family always resolves to a single step, since it has no reveal slot.

Default: `auto`

3.1.11 slide-takeaway

Family 09: dark takeaway slide — a headline and subtext next to a boxed key-takeaway card, on the dark canvas. Use it for the closing synthesis; per the composition rule, open dark (`slide-cover`), close dark with this family, and keep every other family on the light canvas in between. It sets page-level rules internally, so it cannot be called from inside another container or from a live doc example; see the usage snippet below.

```
#slide-takeaway(
  title: "TOPIC",
  section: "NEXT STEPS",
  counter: "06 / 06",
  hero: "Close with the single\nsentence they should remember.",
  subtext: "One line about what happens next.",
  takeaway-title: "START NOW",
  takeaway-text: "TYPE: typst watch main.typ",
)
```

3.1.11.1 Parameters

```
slide-takeaway(
  title: str,
  section: str,
  counter: str,
  hero: str,
  subtext: str,
  takeaway-title: str,
  takeaway-text: str,
```

```

    note: none str,
    subslides: auto int
) -> content

```

3.1.11.1.1 title str

Slide title, shown bold at the top-left by `slide-header`.

Default: ""

3.1.11.1.2 section str

Uppercase section tag shown next to the title.

Default: "CONCLUSION"

3.1.11.1.3 counter str

Page counter text shown at the top-right, e.g. "09 / 10". Omitted entirely when empty.

Default: ""

3.1.11.1.4 hero str

Headline shown above `subtext`. "\n" starts a new line.

Default: ""

3.1.11.1.5 subtext str

Supporting sentence shown under `hero`, in the muted tone.

Default: ""

3.1.11.1.6 takeaway-title str

Uppercase kicker shown above `takeaway-text` inside the boxed card.

Default: "KEY TAKEAWAY"

3.1.11.1.7 takeaway-text str

Bold text shown inside the boxed card.

Default: ""

3.1.11.1.8 note none or str

Speaker note exported to the pdfpc sidecar for this slide. Omitted entirely when `none`.

Default: `none`

3.1.11.1.9 subslides auto or int

Number of subslides (steps) to render; with `auto` this family always resolves to a single step, since it has no reveal slot.

Default: `auto`

3.1.12 slide-three-column

Family 05: three-column mechanism slide (Model | Code | Idea). Use it to walk a mechanism through its three faces at once: a conceptual column, a code listing, and a dark takeaway card. It sets page-level rules internally, so it cannot be called from inside another container or from a live doc example; see the usage snippet below.

```
#slide-three-column(
  title: "TOPIC",
  section: "MECHANISM",
  counter: "04 / 06",
  column1-title: "THE CONCEPT",
  column1-content: block(
    fill: sapians-white,
    stroke: stroke-light,
    radius: radius-sm,
    inset: 3.5mm,
    height: 100%,
    [Concept headline],
  ),
  column1-caption: "One-line caption.",
  column2-title: "THE CODE",
  column2-code: [
    #raw("#slide-cover(title: \"SAPIANS\")", lang: "typst", block: true)
  ],
  column3-title: "THE TAKEAWAY",
  column3-hero: "The point, in 3 words.",
  column3-sub: "One sentence connecting concept and code.",
  column3-footer: "Footer annotation.",
)
```

3.1.12.1 Parameters

```
slide-three-column(
  title: str,
  section: str,
  counter: str,
  column1-title: str,
  column1-content: none content,
  column1-caption: none content,
  column2-title: str,
  column2-code: none content,
  column3-title: str,
  column3-hero: str,
  column3-sub: str,
  column3-footer: str,
  note: none str,
  subslides: auto int
) -> content
```

3.1.12.1.1 title str

Slide title, shown bold at the top-left by `slide-header`.

Default: ""

3.1.12.1.2 section str

Uppercase section tag shown next to the title.

Default: "HOW IT WORKS"

3.1.12.1.3 counter str

Page counter text shown at the top-right, e.g. "05 / 10". Omitted entirely when empty.

Default: ""

3.1.12.1.4 column1-title `str`

Uppercase label above the first column.

Default: "THE MODEL"

3.1.12.1.5 column1-content `none` or `content`

Content of the first column (diagram, card, ...), given a 38mm height. Omitted entirely when `none`.

Default: `none`

3.1.12.1.6 column1-caption `none` or `content`

Small caption shown under the first column's content, in the muted tone. Omitted entirely when `none`.

Default: `none`

3.1.12.1.7 column2-title `str`

Uppercase label above the second column.

Default: "THE CODE"

3.1.12.1.8 column2-code `none` or `content`

Code (or any content) shown inside the second column's `code-box`. Omitted entirely when `none`.

Default: `none`

3.1.12.1.9 column3-title `str`

Uppercase label above the third column, also used as the dark card's kicker.

Default: "THE IDEA"

3.1.12.1.10 column3-hero `str`

Headline shown inside the third column's dark card.

Default: ""

3.1.12.1.11 column3-sub `str`

Supporting sentence shown under `column3-hero`, in the muted tone.

Default: ""

3.1.12.1.12 column3-footer `str`

Bold footer line shown at the bottom of the third column's card.

Default: ""

3.1.12.1.13 note `none` or `str`

Speaker note exported to the pdfpc sidecar for this slide. Omitted entirely when `none`.

Default: **none**

3.1.12.1.14 subslides **auto** or **int**

Number of subslides (steps) to render; with **auto** this family always resolves to a single step, since it has no reveal slot.

Default: **auto**

4 Layouts

The two full-page templates that set up a document from scratch: the two-column scientific article and the single-column technical report. Both set page-level rules internally, so — unlike the components above — their usage is shown as a static snippet rather than a live-rendered example; see each function's description for why.

4.1 Scientific Article

4.1.1 sapians-article

The two-column scientific-article page layout: sets the A4 page, base typography, and a centered title block (kicker, title, authors), then lays the body out in two justified columns. Apply it once per document with a `show` rule — it sets page-level rules, so it cannot be called from inside another container (a card, a box, ...) or from a live doc example; see the usage snippet below.

```
#import "@local/sapians:1.0.0": * // x-release-please-version
```

```
#show: sapians-article.with(
  title: "A Fixed Paper Title",
  abstract: [A short summary of the contribution.],
  authors: (
    (name: "First Author", affiliation: "Institution One"),
    (name: "Second Author", affiliation: "Institution Two"),
  ),
  keywords: ("alpha", "beta", "gamma"),
)
```

= Introduction

Body text flows in two columns from here on.

4.1.1.1 Parameters

```
sapians-article(
  title: str,
  abstract: none | str | content,
  authors: array,
  keywords: array,
  lang: str,
  kicker: str,
  journal: str,
  abstract-title: str,
  keywords-title: str,
  body: content
) -> content
```

4.1.1.1.1 title **str**

Document and title-block title.

Default: "SAPIANS Scientific Article"

4.1.1.1.2 abstract **none** or **str** or **content**

Abstract shown in a bordered box under the title. The abstract block (and keywords, if any) is omitted entirely when **none**.

Default: **none**

4.1.1.1.3 authors `array`

Authors, each either a plain name or a dictionary with `name` and optional `affiliation` keys, e.g. (`name: "Ada", affiliation: "MIT"`). Rendered as one column per author under the title.

Default: `()`

4.1.1.1.4 keywords `array`

Keywords listed at the end of the abstract box, joined with commas. Ignored when `abstract` is `none`.

Default: `()`

4.1.1.1.5 lang `str`

Document language, forwarded to `set text(lang: ...)`.

Default: `"en"`

4.1.1.1.6 kicker `str`

Uppercase kicker shown above the title, e.g. "SAPIANS RESEARCH ARTICLE".

Default: `"SAPIANS RESEARCH ARTICLE"`

4.1.1.1.7 journal `str`

Venue name shown in the page footer.

Default: `"SAPIANS Journal of Applied AI & Design"`

4.1.1.1.8 abstract-title `str`

Label shown before the abstract text.

Default: `"Abstract"`

4.1.1.1.9 keywords-title `str`

Label shown before the keyword list.

Default: `"Keywords"`

4.1.1.1.10 body `content`

Article body, laid out in two columns. Supplied automatically by `#show: sapians-article.with(...)`.

4.2 Technical Report

4.2.1 sapians-report

The single-column technical-report page layout: sets the A4 page (with a running header from the second page on), base typography, and a title block (kicker, title, subtitle, author/date/version line). Apply it once per document with a `show` rule — it sets page-level rules, so it cannot be called from inside another container (a card, a box, ...) or from a live doc example; see the usage snippet below.

```
#import "@local/sapians:1.0.0": * // x-release-please-version

#show: sapians-report.with(
  title: "Fixed Report Title",
```

```

subtitle: "A fixed subtitle for the reference render",
author: "Test Author",
date: "01/01/2026",
version: "1.0",
)

```

= Executive Summary

Body text flows in a single column from here on.

4.2.1.1 Parameters

```

sapians-report(
  title: str,
  subtitle: none | str,
  author: str,
  date: none | str,
  version: str,
  lang: str,
  kicker: str,
  org: str,
  author-title: str,
  date-title: str,
  version-title: str,
  body: content
) -> content

```

4.2.1.1.1 title str

Document and title-block title. Also shown in the running header from the second page on.

Default: "SAPIANS Technical Report"

4.2.1.1.2 subtitle none or str

Subtitle shown under the title. Omitted entirely when none.

Default: none

4.2.1.1.3 author str

Author name shown in the byline row.

Default: "SAPIANS Team"

4.2.1.1.4 date none or str

Date shown in the byline row and the running header, already formatted (e.g. "01/01/2026"). Defaults to today's date as "[day]/[month]/[year]" when none.

Default: none

4.2.1.1.5 version str

Version string shown in the byline row.

Default: "1.0"

4.2.1.1.6 lang str

Document language, forwarded to `set text(lang: ...)`.

Default: "en"

4.2.1.1.7 kicker `str`

Uppercase kicker shown above the title, e.g. "SAPIANS TECHNICAL REPORT".

Default: "SAPIANS TECHNICAL REPORT"

4.2.1.1.8 org `str`

Organization name shown in the page footer.

Default: "SAPIANS Research & DeveLopment"

4.2.1.1.9 author-title `str`

Label shown before the author name in the byline row.

Default: "Author"

4.2.1.1.10 date-title `str`

Label shown before the date in the byline row.

Default: "Date"

4.2.1.1.11 version-title `str`

Label shown before the version string in the byline row.

Default: "Version"

4.2.1.1.12 body `content`

Report body, laid out in a single column. Supplied automatically by `#show: sapians-report.with(...)`.