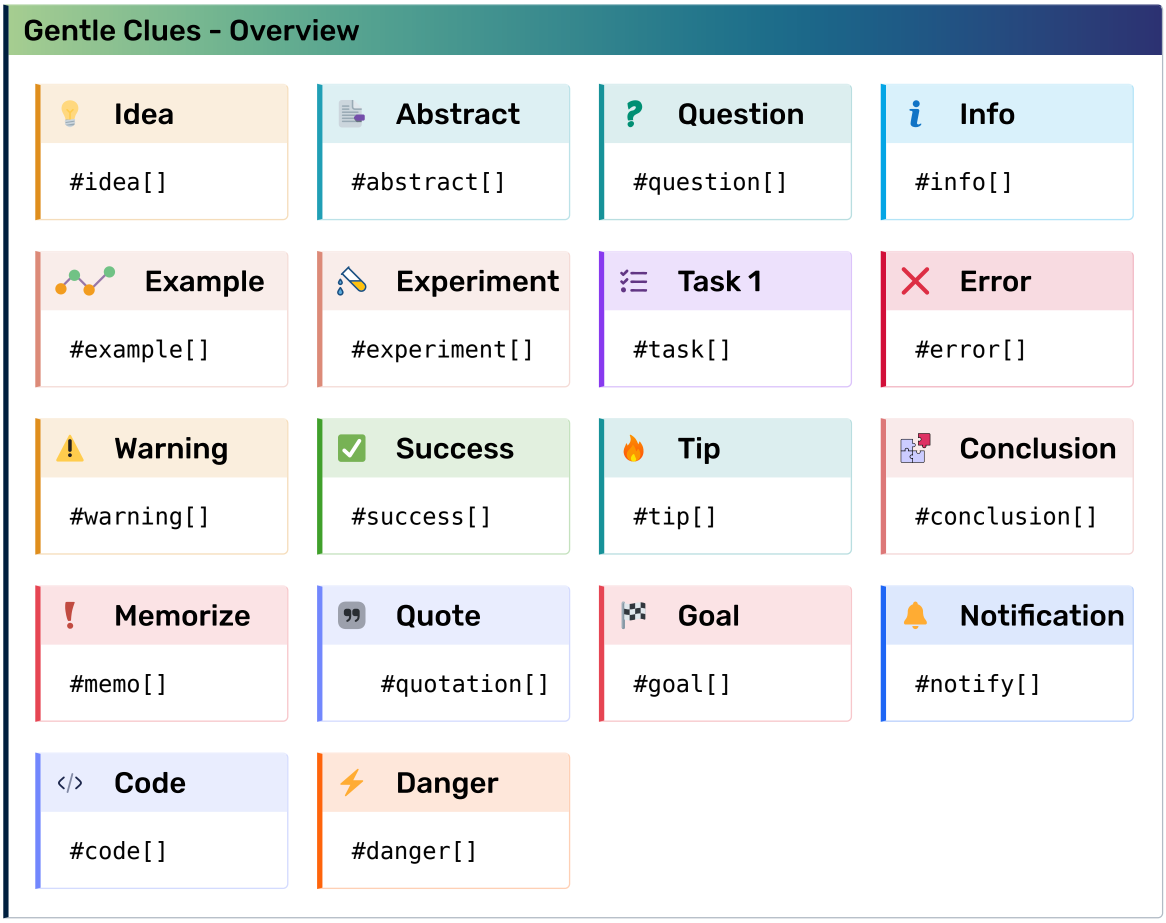
Gentle clues

Add some beautiful, predefined admonitions or define your own.

`gentle-clues` is a package for the typst ecosystem
by jomaway.



Gentle Clues - Overview

Idea #idea[]	Abstract #abstract[]	Question #question[]	Info #info[]
Example #example[]	Experiment #experiment[]	Task 1 #task[]	Error #error[]
Warning #warning[]	Success #success[]	Tip #tip[]	Conclusion #conclusion[]
Memorize #memo[]	Quote #quotation[]	Goal #goal[]	Notification #notify[]
Code #code[]	Danger #danger[]		

Figure 1: Overview of all predefined clues.

Contents

1. Getting started

Minimal starting example

```
#import "@preview/gentle-clues:1.1.0": *
#tip[Check out this cool package]
```

Turns into this ⟶

> 🔥 **Tip**
>
> Check out this cool package

Usage

1. Import the package like this:

   ```
   #import "@preview/gentle-clues:1.1.0": *
   ```

2. **Optional:** Change the default settings for all clues if desired. See Example 2.

3. Use a predefined clue without any options

   ```
   #info[You will find a list with all
   predefined clues at the last page.]
   ```

   Turns into this ⟶

   > *i* **Info**
   >
   > You will find a list with all predefined clues at the last page.

4. Or overwrite the default parameters. e.g. set a custom title

   ```
   #example(title: "Custom title")
   [ Content ...]
   ```

   Turns into this ⟶

   > **Custom title**
   >
   > Content …

   See **all available parameters** at Listing 1.

5. **I18n:**
   - The current language which is set by `#set text(lang: "de")` changes the default header title.
   - Currently supported are “en” , “de” , “es” , “fr” , “hu” , “zh” and “it” . This package uses linguify for language settings. Feel free to contribute more languages.

2. Features

> 🔔 **Breaking news**
>
> Clues can now break onto the next page with option: `breakable: true`
>
> Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat voluptatem. Ut enim aeque doleamus animo, cum corpore dolemus, fieri.

This is a two columns layout.

Color management and title font

Clues can be styled in your liking.

The simplest way is to change the `accent-color` which will be the thick border stroke on the left side. Header and border color will then automatically derived from this color.

But you can set the `header-color`,`border-color` and `body-color` independently with a `color`, `gradient` or `pattern`.

Additionally, you can set a different font for the title with `title-font` and its weight offset with `title-weight-delta`.

Example:

```
#clue(
  title: "Color management and title font",
  accent-color: gradient.linear(red, blue, dir:ttb),
  header-color: gradient.linear(red, yellow, blue),
  border-color: blue.darken(40%),
  body-color: pattern(text(fill:fuchsia.lighten(80%)," . ")),
  title-font: "Liberation Mono",
  title-weight-delta: 300
)[...]
```

Define your own clue

```
// Define a clue called ghost
#let ghost(title: "Buuuuuuh", icon: emoji.ghost , ..args) = clue(
  accent-color: purple, // Define a base color
  title: title,    // Define the default title
  icon: icon,      // Define the default icon
  ..args           // Pass along all other arguments
)
// Use it
#ghost[Huuuuuuh.]
```

The result looks like this.

👻 **Buuuuuuh.**

Huuuuuuh.

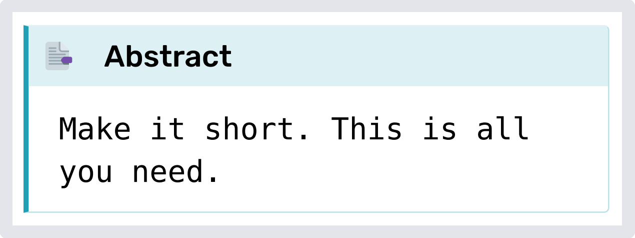
List of all predefined clues

3. Predefined Clues

- abstract()
- code()
- conclusion()
- danger()
- error()
- example()
- experiment()
- goal()
- idea()
- info()
- memo()
- notify()
- question()
- quotation()
- success()
- task()
- tip()
- warning()

abstract

```
#abstract[Make it short. This is all you
need.]
```



Abstract

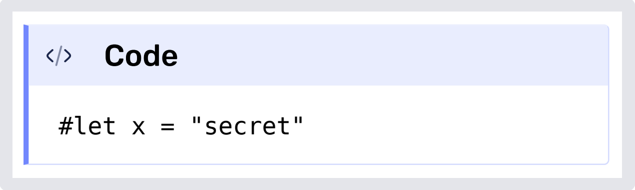
Make it short. This is all you need.

Parameters

```
abstract(..args) -> content
```

code

```
#code[`#let x = "secret"`]
```



Code

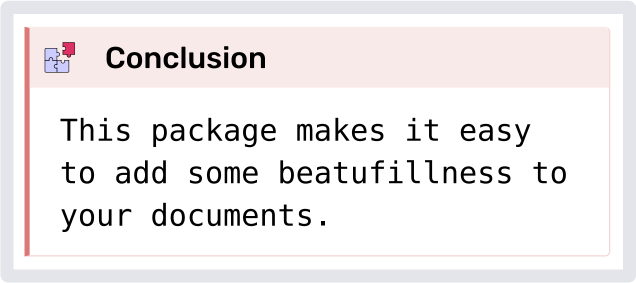
#let x = "secret"

Parameters

```
code(..args) -> content
```

conclusion

```
#conclusion[This package makes it easy to
add some beatufillness to your documents.]
```



Conclusion

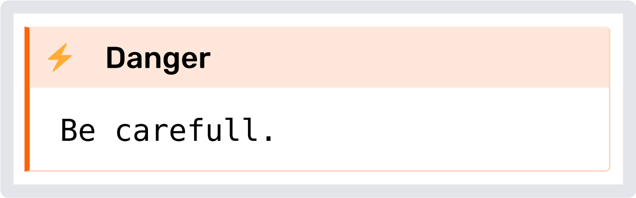
This package makes it easy to add some beatufillness to your documents.

Parameters

```
conclusion(..args) -> content
```

danger

```
#danger[Be carefull.]
```



Danger

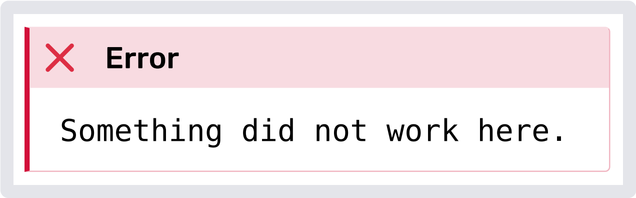
Be carefull.

Parameters

```
danger(..args) -> content
```

error

```
#error[Something did not work here.]
```



Error

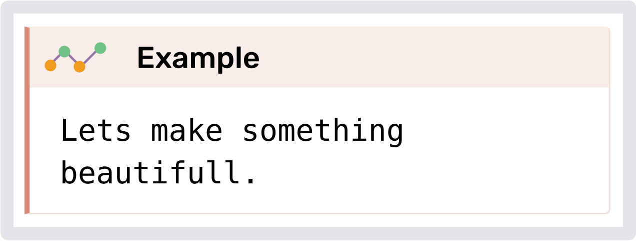
Something did not work here.

Parameters

```
error(..args) -> content
```

example

```
#example[Lets make something beautifull.]
```



Example

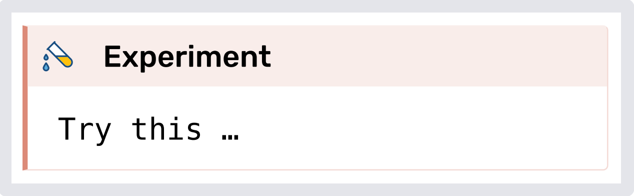
Lets make something beautifull.

Parameters

```
example(..args) -> content
```

experiment

```
#experiment[Try this ...]
```



Experiment

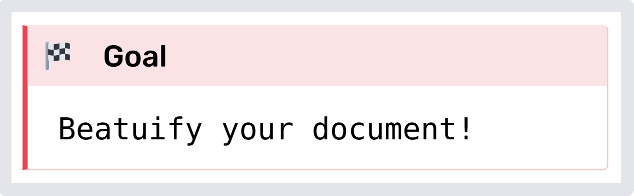
Try this …

Parameters

```
experiment(..args) -> content
```

goal

```
#goal[Beatuify your document!]
```



Goal

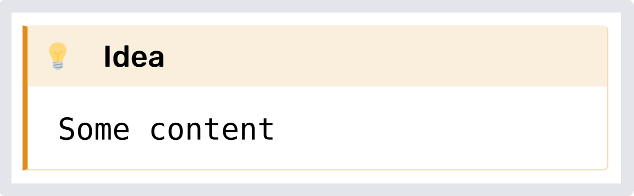
Beatuify your document!

Parameters

```
goal(..args) -> content
```

idea

```
#idea[Some content]
```



Idea

Some content

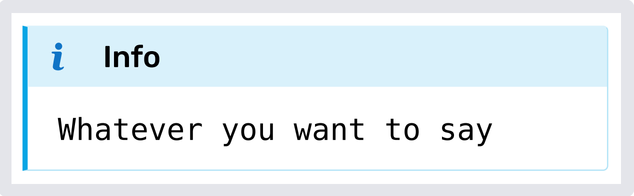
Parameters

```
idea(..args) -> content
```

info

Info

```
#info[Whatever you want to say]
```



Info

Whatever you want to say

Parameters

```
info(..args) -> content
```

memo

```
#memo[Leave a #emoji.star on github.]
```

> ❗ **Memorize**
>
> Leave a ⭐ on github.

Parameters

```
memo(..args) -> content
```

notify

Notificaton

```
#notify[New features in future versions.]
```

> 🔔 **Notification**
>
> New features in future versions.

Parameters

```
notify(..args) -> content
```

question

```
#question[How do amonishments work?]
```

> ❓ **Question**
>
> How do amonishments work?

Parameters

```
question(..args) -> content
```

quotation

```
#quotation(attribution: "The maintainer")
[Keep it simple. Admonish your life.]
```

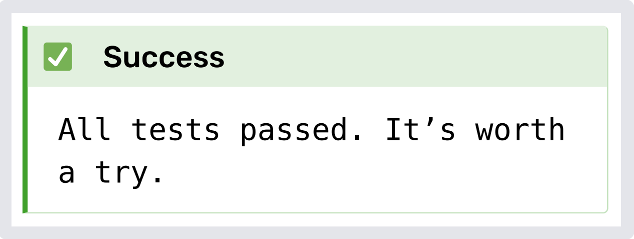
> ❞ **Quote**
>
> Keep it simple. Admonish your life.
>
> — The maintainer

Parameters

```
quotation(
  attribution,
  content,
  ..args
) -> content
```

success

```
#success[All tests passed. It's worth a
try.]
```



☑ **Success**

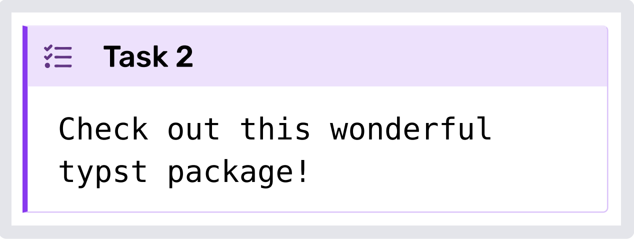
All tests passed. It’s worth a try.

Parameters

```
success(..args) -> content
```

task

```
#task[Check out this wonderful typst
package!]
```



Task 2

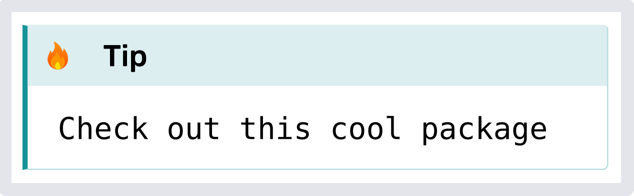
Check out this wonderful typst package!

Parameters

```
task(..args) -> content
```

tip

```
#tip[Check out this cool package]
```



🔥 **Tip**

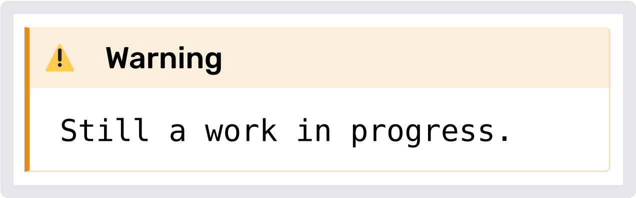
Check out this cool package

Parameters

```
tip(..args) -> content
```

warning

```
#warning[Still a work in progress.]
```



⚠ **Warning**

Still a work in progress.

Parameters

```
warning(..args) -> content
```

4. Gentle Clues API

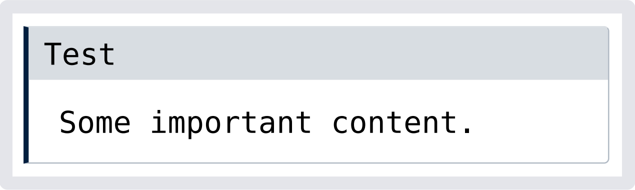
clue

Basic gentle-clue (clue) template function.

This function can be used to create custom clues. You can pass all of this parameters to the predefined clues (Section) as well to overwrite the default settings. If an argument is `auto` it will fallback to the value which is specified in `gentle-clues()`

Example:

```
#clue(title:"Test")[Some important content.]
```



Test

Some important content.

Parameters

```
clue(
  content: content,
  title: string none,
  icon: none image symbol,
  accent-color: color gradient pattern,
  header-color: auto color gradient pattern,
  border-color: auto color gradient pattern,
  body-color: none color gradient pattern,
  width: auto length,
  radius: auto length,
  border-width: auto length,
  stroke-width: auto length,
  content-inset: auto length,
  header-inset: auto length,
  title-font: auto string,
  title-weight-delta: auto int,
  breakable: auto boolean,
  headless: boolean
) -> content
```

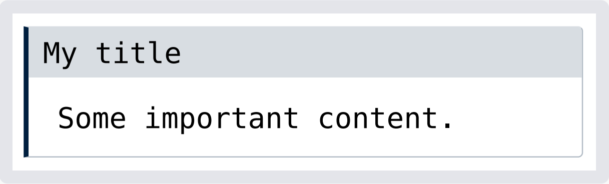
content `content`

Content inside the body.

title `string` or `none`

The title of the clue. If `none` the header is ommited.

```
#clue(title:"My title")[Some important
content.]
```



My title

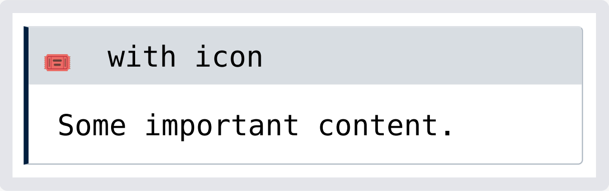
Some important content.

Default: `none`

icon `none` or `image` or `symbol`

The icon to show in front of the title.

```
#clue(title:"with icon", icon:
emoji.ticket)[Some important content.]
```



with icon

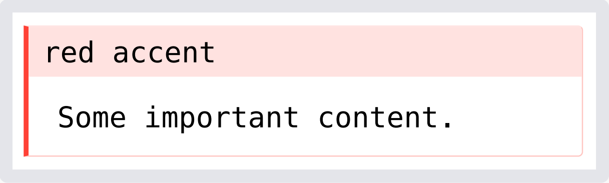
Some important content.

Default: `none`

accent-color `color` or `gradient` or `pattern`

The accent color defines the color of the stroke on the left side. If border-color and header-color are set to auto those get derived from the accent color.

```
#clue(title:"red accent", accent-color:
red)[Some important content.]
```



red accent

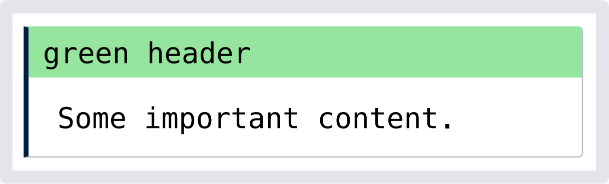
Some important content.

Default: `navy`

header-color `auto` or `color` or `gradient` or `pattern`

The background color of the header.

```
#clue(title:"green header", header-color:
green.lighten(50%))[Some important
content.]
```



green header

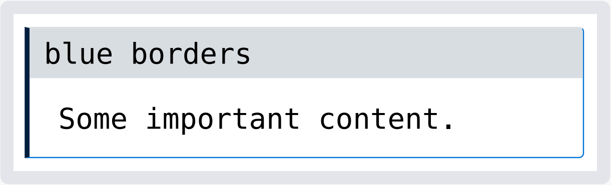
Some important content.

Default: `auto`

border-color `auto` or `color` or `gradient` or `pattern`

The color of the small border on the bottom and right side. if in headless mode also for the top border.

```
#clue(title: "blue borders", border-color:
blue)[Some important content.]
```



blue borders

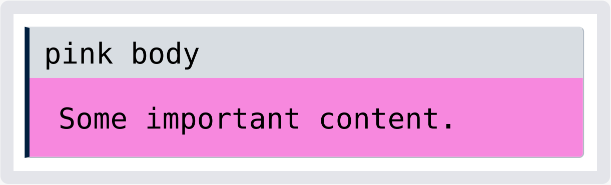
Some important content.

Default: `auto`

body-color `none` or `color` or `gradient` or `pattern`

The background color of the body

```
#clue(title:"pink body", body-color:
fuchsia.lighten(50%))[Some important
content.]
```



pink body

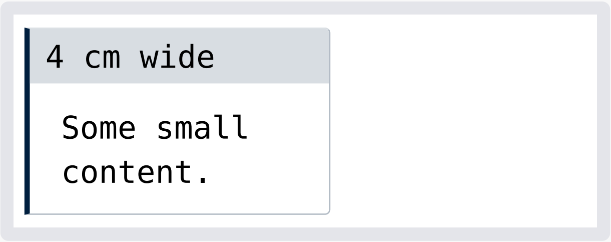
Some important content.

Default: `none`

width `auto` or `length`

The total width of the clue.

```
#clue(title:"4 cm wide", width: 4cm)[Some
small content.]
```



4 cm wide

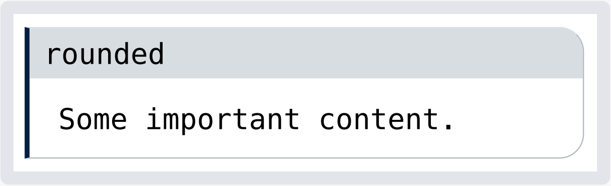
Some small content.

Default: `auto`

radius `auto` or `length`

The border radius on the right side.

```
#clue(title:"rounded", radius: 10pt)[Some
important content.]
```



rounded

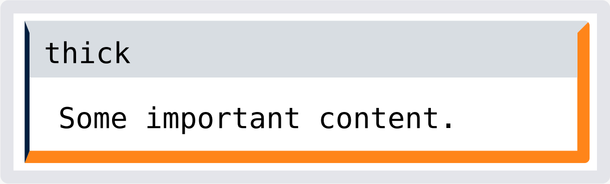
Some important content.

Default: `auto`

border-width `auto` or `length`

Defines the width of the border.

```
#clue(title:"thick", border-width: 5pt,
border-color: orange)[Some important
content.]
```



thick

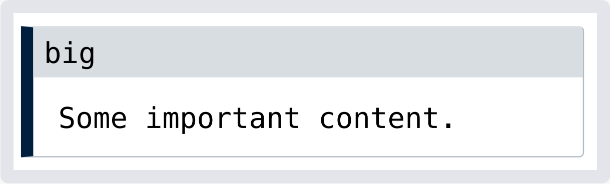
Some important content.

Default: `auto`

stroke-width `auto` or `length`

Defines the width of the stroke on the left side.

```
#clue(title:"big", stroke-width: 5pt,)
[Some important content.]
```



big

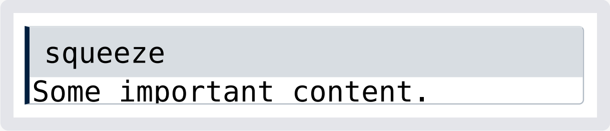
Some important content.

Default: `auto`

content-inset `auto` or `length`

The inset for the content body.

```
#clue(title:"squeeze", content-inset: 1pt)
[Some important content.]
```



squeeze

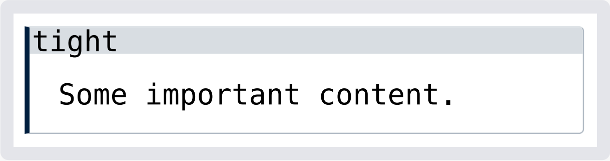
Some important content.

Default: `auto`

header-inset `auto` or `length`

Defines the inset for the header

```
#clue(title:"tight", header-inset: 1pt)
[Some important content.]
```



tight

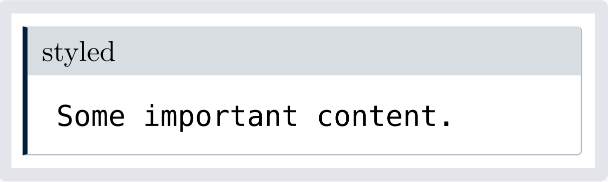
Some important content.

Default: `auto`

title-font `auto` or `string`

The font which is used for the title.

```
#clue(title:"styled", title-font:"New
Computer Modern" )[Some important
content.]
```



styled

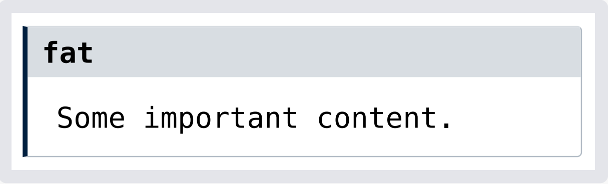
Some important content.

Default: `auto`

title-weight-delta `auto` or `int`

The weight delta which is used for the title. Output depends on the used font.

```
#clue(title:"fat", title-weight-delta:
500)[Some important content.]
```



fat

Some important content.

Default: `auto`

breakable `auto` or `boolean`

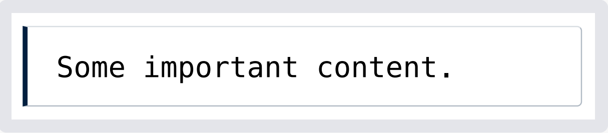
Defines if the clue is breakable. If `auto` it falls back to the default settings. See `gentle-clues()`

Default: `auto`

headless `boolean`

Defines if clues should be shown without a header. If `auto` it falls back to the default settings. See `gentle-clues()`

```
#clue(title:"fat", headless: true)[Some
important content.]
```



Some important content.

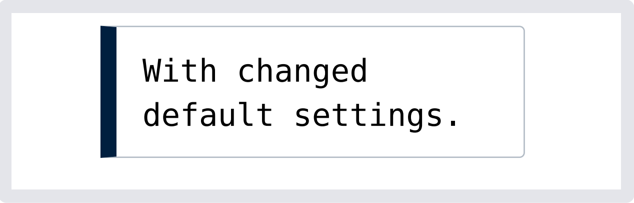
Default: `auto`

gentle-clues

Config the default settings for all clues globally.

Example: Change the default settings for all clues.

```
#show: gentle-clues.with(
  headless: true,
  stroke-width: 6pt,
  width: 5.5cm,
)
#clue[With changed default settings.]
```



With changed default settings.

Parameters

```
gentle-clues(
  breakable: boolean,
  headless: boolean,
  header-inset: length,
  width: auto length,
  stroke-width: length,
  border-radius: length,
  border-width: length,
  content-inset: length,
  title-font: auto string,
  title-weight-delta: int,
  body
) -> content
```

breakable boolean

defines if clues break across pages by default.

Default: `false`

headless boolean

defines if clues should be shown without a header by default.

Default: `false`

header-inset length

defines the default header-inset for all clues.

Default: `0.5em`

width auto or length

defines the default width for all clues.

Default: `auto`

stroke-width length

defines the default stroke width of the left colored stroke for all clues.

Default: `2pt`

border-radius `length`

defines the default border radius for all clues.

Default: `2pt`

border-width `length`

defines the default border width for all clues.

Default: `0.5pt`

content-inset `length`

defines the default content inset of the body for all clues.

Default: `1em`

title-font `auto` or `string`

defines the default font for the title

Default: `auto`

title-weight-delta `int`

defines the default weight offset for the title

Default: `200`