

Gentle clues

Add some beautiful, predefined admonitions or define your own.

gentle-clues is a package for the typst ecosystem
by [jomaway](#).

Version: 1.3.0

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

Figure 1: Overview of all predefined clues.

Contents

Gentle clues	1
1. Getting started	2
2. Features	2
3. Define your own clues	3
4. Predefined Clues	4
5. Gentle Clues API	12

1. Getting started

Minimal starting example

```
#import "@preview/gentle-clues:1.3.0": *
```

`#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:1.3.0": *
```

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 →



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 API 1.

5. **I18n:**

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

2. Features

This package provides some features which helps to customize the clues to your liking.

- Brings a wide variety of predefined clues. See Section 2.
- Set global default for all clues. See Example 2.
- Overwrite each parameter on a single clue for changing title, color, etc. See Section 1.
- Show or hide a counter value on tasks.
- Define your own clues very easily. See Section 3.

2.1. Custom styling

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: "Rainbow style",  
  accent-color: gradient.linear(red,  
blue, dir: ttb),  
  header-color: gradient.linear(red,  
yellow, blue),  
  border-color: blue.darken(40%),  
  body-color: yellow.lighten(80%),  
  title-font: "Liberation Mono",  
  title-weight-delta: 300  
)  
[Some content. #lorem(20)]
```

Rainbow style

Some content. Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua quaerat.

There are many more options for customization. For all possible parameters see API 1.

3. Define your own clues

You can easily define your own clues. Just set some default values for `color`, `title`, `icon`, ... and you are ready to go.

```
// Define a clue called ghost  
#let ghost(title: "Buuuuuuh", icon:  
emoji.ghost, ...args) = clue(  
  // Define default values.  
  accent-color: purple,  
  title: title,  
  icon: icon,  
  // Pass along all other arguments  
  ...args  
)  
  
// Use it  
#ghost[This is the ghost number one.]  
#ghost(title: "Poltergeist")[This ghost  
has a custom name.]  
#ghost[Huuuuuuh.]
```

Buuuuuuuh

This is the ghost number one.

Poltergeist

This ghost has a custom name.

Buuuuuuuh

Huuuuuuh.

4. Predefined Clues

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

4.1. abstract

Color	Icon	Title	Note
#209fb5		Abstract	

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

Abstract

Make it short. This is all you need.

Parameters

..args arguments

Supports all parameters of [clue\(\)](#).

4.2. code

Color	Icon	Title	Note
#7287fd		Code	

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

</> Code

#let x = "secret"

Parameters

4.3. conclusion

Color	Icon	Title	Note
#dd7878		Conclusion	

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

Conclusion

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

Parameters

4.4. danger

Color	Icon	Title	Note
#fe640b		Danger	

`#danger`[Be careful.]

Danger

Be careful.

Parameters

4.5. error

Color	Icon	Title	Note
#d20f39		Error	

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

Error

Something did not work here.

Parameters

4.6. example

Color	Icon	Title	Note
#dc8a78		Example	

```
#example[Let's make something beautiful.]
```

 **Example**

Let's make something beautiful.

Parameters**4.7. experiment**

Color	Icon	Title	Note
#dc8a78		Experiment	

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

 **Experiment**

Try this ...

Parameters**4.8. goal**

Color	Icon	Title	Note
#e64553		Goal	

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

 **Goal**

Beautify your document!

Parameters

4.9. idea

Color	Icon	Title	Note
 #df8e1d		Idea	

#idea[Some content]

 Idea

Some content

Parameters**4.10. info**

Color	Icon	Title	Note
 #04a5e5		Info	

#info[Whatever you want to say]

 Info

Whatever you want to say

Parameters**4.11. memo**

Color	Icon	Title	Note
 #e64553	!	Memorize	

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

! Memorize

Leave a  on github.**Parameters**

4.12. notify

Color	Icon	Title	Note
 #1e66f5		Notification	

Notification

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

 **Notification**

New features in future
versions.

Parameters

4.13. question

Color	Icon	Title	Note
 #179299		Question	

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

 **Question**

How do admonishments work?

Parameters

4.14. quotation

Color	Icon	Title	Note
#7287fd	“ ”	Quote	

```
// With attribution
#quotation(attribution: "The maintainer")
[Keep it simple. Admonish your life.]
// Without attribution
#quotation()[Keep it simple. Admonish your
life.]
```

“ ” Quote

Keep it simple. Admonish
your life.

– The maintainer

“ ” Quote

Keep it simple. Admonish
your life.

Parameters

attribution `string` or `none`

The author of the quote.

- If `none` is given (default), only the quote content will be shown.
- If provided, the quote will be shown as `typst quote` with the attribution passed to it.

Default: `none`

content `content`

the quote itself

..args `arguments`

Supports all parameter from `clue()`.

4.15. success

Color	Icon	Title	Note
 #40a02b		Success	

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

 Success

All tests passed. It's worth a try.

Parameters

4.16. task

Color	Icon	Title	Note
 #8839ef		Task	

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

 Task 2

Check out this wonderful typst package!

Parameters

4.17. tip

Color	Icon	Title	Note
 #179299		Tip	

`#tip`[Check out this cool package]

 Tip

Check out this cool package

Parameters

4.18. warning

Color	Icon	Title	Note
#df8e1d	⚠	Warning	

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

⚠ Warning

Still a work in progress.

Parameters

5. Gentle Clues API

- [clue\(\)](#)
- [gentle-clues\(\)](#)

5.1. clue

Basic gentle-clue (clue) template function.

This function can be used to create custom clues. You can pass all of these parameters to the predefined clues (Section 2) 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.]
```

The screenshot shows a user interface element consisting of two nested rectangular boxes. The inner box is light gray and contains the text "Some important content.". Above it, a darker gray header box contains the text "Test".

Parameters

```
clue(
  content: content,
  title: string | none,
  icon: none | image | symbol,
  accent-color: color | gradient | tiling,
  header-color: auto | color | gradient | tiling,
  border-color: auto | color | gradient | tiling,
  body-color: none | color | gradient | tiling,
  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 omitted.

```
#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 `tiling`

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 `tiling`

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 `tiling`

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 `tiling`

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

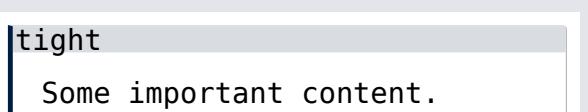
Some important content.

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`

5.2. 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`