# Component Customization Showcase
## Demonstrating Enhanced Components

07/09/2025 - 23/11/2025

**Tom Planche**
2025-2026
*Template Documentation*

# Table des matières

<div align="center">

**Chapter 1**

</div>

---

<div align="center">

# Component Customization Showcase

</div>

---

This document demonstrates the enhanced customization capabilities of all template components.

## I. I - Math Components

All math components now support extensive customization including colors, spacing, borders, and more.

### I. I .I - Standard Definition (Default Styling)

> **Definition 1.1.** (Linéarité): On dit que $\varphi$ est linéaire (homomorphisme) si:
> $$\varphi(\lambda_1 X_1 + \lambda_2 X_2 + ... + \lambda_n X_n) = \lambda_1 \varphi(X_1) + \lambda_2 \varphi(X_2) + ... + \lambda_n \varphi(X_n) \qquad (1.1.1.1)$$

### I. I .II - Custom Colored Definition

> **Definition 1.2.** (Prime Number): A **prime number** is a natural number greater than 1 that has no positive divisors other than 1 and itself.

### I. I .III - Standard Example (Default Styling)

> **Example 1.1.** (Basic Example): Basic text.
> Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magnam aliquam quaerat.
> $$\varphi(0,0,0) = (0,0) = 0_{\mathbb{R}^2}$$
> $$\varphi(\alpha X_1 + \beta X_2) \overset{?}{=} \alpha \varphi(X_1) + \beta \varphi(X_2) \qquad (1.1.3.2)$$

### I. I .IV - Custom Colored Example

> **Example 1.2.** (Factorial Calculation): The factorial of 5 is calculated as:
> $$5! = 5 \times 4 \times 3 \times 2 \times 1 = 120 \qquad (1.1.4.3)$$

### I. I .V - Standard Theorem (Default Styling)

**Theorem 1.1.** (Théorème de Stokes): Soit $M$ une variété différentielle à bord, orientée de dimension $n$, et $\omega$ une $(n-1)$-forme différentielle à support compact sur $M$ de classe $C_1$. Alors, on a :

$$\int_M d\omega = \int_{\{\partial M\}} i^*\omega \tag{1.1.5.4}$$

où $d$ désigne la dérivée extérieure, $\partial M$ le bord de $M$, muni de l'orientation induite, et $i^*\omega = \omega \mid_{\{\partial M\}}$ la restriction de $\omega$ à $\partial M$.

### I. I .VI - Custom Colored Theorem (Breakable)

**Theorem 1.2.** (Pythagorean Theorem): For any right triangle with sides $a$, $b$, and hypotenuse $c$:

$$a^2 + b^2 = c^2 \tag{1.1.6.5}$$

## I. II - Block Components

### I. II .I - Standard `my-block` (Default)

This is a standard block with default styling. It's centered with a light gray background.

### I. II .II - Block with Title (Centered)

**Important Note**

This block has a centered title and custom warning colors.

### I. II .III - Full-Width Block with Right-Aligned Content

**Right-Aligned Content**

This content is aligned to the right within a full-width block.

### I. II .IV - Auto-Width Block (Left-Aligned)

This block fits its content width and is aligned to the left.

## I. III - Blockquote Components

### I. III .I - Standard Blockquote (Default)

This is a standard blockquote with the default left border accent.

### I. III .II - Blockquote with Attribution

Imagination is more important than knowledge. Knowledge is limited. Imagination encircles the world.

— Albert Einstein

### I. III .III - Right-Side Border with Centered Content

I hear and I forget. I see and I remember. I do and I understand.

— Confucius

### I. III .IV - Full Border with Custom Styling

Nothing in life is to be feared, it is only to be understood. Now is the time to understand more, so that we may fear less.

— Marie Curie

### I. III .V - Top Border with Custom Width

Be yourself; everyone else is already taken.

— Oscar Wilde

# I. IV - Code Components

## I. IV .I - Standard Code Block (Default)

```
Python
1  def fibonacci(n):
2      if n ≤ 1:
3          return n
4      return fibonacci(n-1) + fibonacci(n-2)
```

## I. IV .II - Code with Title and Filename

**String Utility Functions**

```
src/string_utils.rs | Rust
 1  /// Extension traits and utilities for string manipulation
 2  pub trait TitleCase {
 3      fn to_title_case(&self) → String;
 4  }
 5
 6  impl TitleCase for str {
 7      fn to_title_case(&self) → String {
 8          self.split(|c: char| c.is_whitespace() || c = '_' || c = '-')
 9              .filter(|s| !s.is_empty())
10              .map(|word| {
11                  let mut chars = word.chars();
12                  match chars.next() {
13                      None ⇒ String::new(),
14                      Some(first) ⇒ {
15                          let first_upper = first.to_uppercase().collect::<String>();
16                          let rest_lower = chars.as_str().to_lowercase();
17                          format!("{}{}", first_upper, rest_lower)
18                      }
19                  }
20              })
21              .collect::<Vec<String>>()
22              .join(" ")
23      }
24  }
```

## I. IV .III - Centered Code Block without Line Numbers

```
TypeScript
interface User {
    id: string;
    name: string;
    email: string;
}

type UserWithoutId = Omit<User, 'id'>;
```

## I. IV .IV - Code with Line Range (Lines 5-15)

**Partial Code View**

```
algorithms.py | Python
```

```
 5      pivot = arr[len(arr) // 2]
 6      left = [x for x in arr if x < pivot]
 7      middle = [x for x in arr if x == pivot]
 8      right = [x for x in arr if x > pivot]
 9
10      return quicksort(left) + middle + quicksort(right)
11
12  def binary_search(arr, target):
13      left, right = 0, len(arr) - 1
14
15      while left <= right:
```

## I. V - Vector Notation

For vectors, use `ar(X)` to get $\vec{X}$.

# Summary of Customization Options

This template now provides extensive customization for all components:

### Math Components

- **definition**, **example**, **theorem**
- Customize: fill, stroke, radius, inset, numbering, breakable
- Full control over colors and spacing

### Block Components

- **my-block**
- Features: title, alignment (block & content), width control
- Perfect for callouts, notes, and highlighted content

### Blockquote Components

- **blockquote**
- Features: attribution, border positioning, alignment
- Support for all four sides or full border

### Code Components

- **code**
- Features: title, line numbers, line ranges, custom styling
- Extensive options for professional code display