



Utiliser Codly dans Typst

publié le 01/03/2026

Pourquoi les enseignants de NSI et SNT devraient l'utiliser

Descriptif :

Codly est un package qui optimise les blocs de code pour vos documents Typst. Il vous permet d'ajouter des annotations, de sauter des lignes, de personnaliser la numérotation, d'ajouter des icônes de langue et bien plus encore.

Sommaire :

- Codly transforme un bloc de code en outil pédagogique
 - Exemples en python
 - Exemples en SQL
 - Exemples en HTML, CSS et JavaScript
-

Codly est un package qui optimise les blocs de code pour vos documents Typst. Il vous permet d'ajouter des annotations, de sauter des lignes, de personnaliser la numérotation, d'ajouter des icônes de langue et bien plus encore.

Cet article a utilisé comme référence le site de la [documentation de Codly](#)

● Codly transforme un bloc de code en outil pédagogique

Produire des supports clairs est devenu une compétence pédagogique à part entière. En NSI comme en SNT, nous enseignons avec du code, mais aussi à travers le code. Codly est là pour cela.

L'intérêt de Codly n'est pas seulement esthétique — il est didactique.

○ Une lisibilité immédiatement supérieure

Codly améliore automatiquement :

- l'espacement
- la hiérarchie visuelle
- la séparation des blocs
- le contraste

Résultat : ☑ le cerveau identifie plus vite les structures.

Les élèves repèrent mieux :

- boucles
- conditions
- fonctions
- indentations

○ Les numéros de ligne : un levier pédagogique majeur

Grace aux numéros, on peut identifier des lignes ou des blocs et y faire référence.

○ Mettre en évidence une ligne clé

Codly permet de surligner certaines lignes.

C'est idéal pour :

- montrer une erreur classique
- attirer l'attention sur une condition
- expliquer un invariant de boucle
- détailler un algorithme

Le code devient alors un support interactif, presque narratif.

● Exemples en python

```
#import "@preview/codly:1.3.0": *
#import "@preview/codly-languages:0.1.1": *
#show: codly-init.with()

== Ajout de surlignages
#codly(highlights: (
  (line: 4, start: 3, end: none, fill: red),
  (line: 5, start: 15, end: 21, fill: green),
  (line: 5, start: 25, fill: blue),
))
```python
def fib(n):
 if n <= 1:
 return n
 else:
 return fib(n - 1) + fib(n - 2)
print(fib(25))
```

== Ajout d'annotations
#codly(
  annotations: (
    (
      start: 1,
      end: 5,
      content: block(
        width: 2em,
        rotate(-90deg, reflow: true,
          align(center, box(width: 100pt)[corps de fonction])
        )
      )
    ),
  )
)
```py
def fib(n):
 if n <= 1:
 return n
 else:
 return fib(n - 1) + fib(n - 2)
```

```
print(fib(25))
```

```
...
```

#### Ajout de surlignages

```
1 def fib(n): python
2 if n <= 1:
3 return n
4 else:
5 return fib(n - 1) + fib(n - 2)
6 print(fib(25))
```

#### Ajout d'annotations

```
1 def fib(n): py
2 if n <= 1:
3 return n
4 else:
5 return fib(n - 1) + fib(n - 2)
6 print(fib(25))
```

} (1)  
corps de fonction

### ● Exemples en SQL

```
#import "@preview/codly:1.3.0": *
#import "@preview/codly-languages:0.1.1": *
#show: codly-init.with()
```

== Exemple simple en SQL

```
```sql
SELECT e.nom, m.nom AS matiere, r.note
FROM Resultats r
JOIN Eleves e ON r.eleve_id = e.id
JOIN Matieres m ON r.matiere_id = m.id
WHERE m.nom = 'NSI';
]
```

```
...
```

== Exemple avec mise en avant

```
#codly(highlights:(
(line: 3, start: 0, fill: blue),
))
```sql
SELECT nom, prenom
FROM Eleves
WHERE moyenne > 20;
...
```

#### Exemple simple en SQL

```
1 SELECT e.nom, m.nom AS matiere, r.note sql
2 FROM Resultats r
3 JOIN Eleves e ON r.eleve_id = e.id
4 JOIN Matieres m ON r.matiere_id = m.id
5 WHERE m.nom = 'NSI';
6]
```

#### Exemple avec mise en avant

```
1 SELECT nom, prenom sql
2 FROM Eleves
3 WHERE moyenne > 20;
```

### ● Exemples en HTML, CSS et JavaScript

```
#import "@preview/codly:1.3.0": *
#import "@preview/codly-languages:0.1.1": *
#show: codly-init.with()
```

== Codly avec du code HTML :

```
```html
<!DOCTYPE html>
<html>
```

```
<head>
  <meta charset="UTF-8">
  <title>Changer la couleur</title>
  <link rel="stylesheet" href="style.css">
</head>

<body>
  <button onclick="changerCouleur()">
    Cliquez-moi !
  </button>

  <script src="script.js"></script>
</body>
</html>
]
```

```
...
```

== Codly avec du code CSS :

```
```css
body {
 background-color: white;
 text-align: center;
 margin-top: 100px;
}

button {
 padding: 12px 20px;
 font-size: 18px;
 cursor: pointer;
}
]
```
```

== Codly avec du code JavaScript :

```
```js
function changerCouleur() {
 document.body.style.backgroundColor =
 document.body.style.backgroundColor === "lightblue"
 ? "white"
 : "lightblue";
}
]
```
```

Codly avec du code HTML :

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <title>Changer la couleur</title>
6   <link rel="stylesheet" href="style.css">
7 </head>
8
9 <body>
10  <button onclick="changerCouleur()">
11    Cliquez-moi !
12  </button>
13
14  <script src="script.js"></script>
15 </body>
16 </html>
17 ]
```

Codly avec du code CSS :

```
1 body {
2   background-color: white;
3   text-align: center;
4   margin-top: 100px;
5 }
6
7 button {
8   padding: 12px 20px;
9   font-size: 18px;
10  cursor: pointer;
11 }
12 ]
```

Codly avec du code JavaScript :

```
1 function changerCouleur() {
2   document.body.style.backgroundColor =
3     document.body.style.backgroundColor === "lightblue"
4     ? "white"
5     : "lightblue";
6 }
7 ]
```