



Université de Lille

Typographie

Projet de typographie avec L<sup>A</sup>T<sub>E</sub>X

Rapport

# Développement

**Étudiants :**

Letudiant Joel

Letudiante Joelle

**Enseignant :**

Dr. LaTeX

1<sup>er</sup> août 2025

# Table des matières

<b>1</b>	<b>Introduction et Concepts de Base</b>	<b>2</b>
1.1	Listes à puces et numérotées . . . . .	2
1.2	Structure des Titres . . . . .	3
1.2.1	Ceci est un sous-sous-titre . . . . .	3
<b>2</b>	<b>Mathématiques et Environnements Scientifiques</b>	<b>4</b>
2.1	Équations et Formules . . . . .	4
2.2	Théorèmes, Définitions et Remarques . . . . .	4
2.3	Unités Scientifiques . . . . .	4
<b>3</b>	<b>Éléments Visuels : Figures et Tableaux</b>	<b>6</b>
3.1	Insertion de Figures . . . . .	6
3.2	Création de Tableaux . . . . .	6
<b>4</b>	<b>Boîtes d’Information et Listings de Code</b>	<b>8</b>
4.1	Boîtes d’Information . . . . .	8
4.2	Listings de Code . . . . .	8
<b>5</b>	<b>Nomenclature et Références</b>	<b>11</b>
5.1	Nomenclature . . . . .	11
5.2	Références et Hyperliens . . . . .	11
<b>6</b>	<b>Conclusion</b>	<b>12</b>

# 1 Introduction et Concepts de Base

Ce document a pour objectif de démontrer les multiples fonctionnalités offertes par la classe  $\text{\LaTeX}$  personnalisée `rapport.cls`. Chaque section explorera un ensemble de commandes et d'environnements spécifiques pour illustrer leur utilisation et leur rendu visuel.

## Information

Ce document est auto-descriptif. Le code  $\text{\LaTeX}$  qui le génère est un exemple direct de l'utilisation de la classe. N'hésitez pas à consulter le fichier `.tex` pour voir comment chaque élément est implémenté.

Nous allons commencer par les éléments de base tels que les listes et les différents niveaux de titres, avant de passer à des sujets plus complexes. L'utilisation correcte des guillemets typographiques français se fait avec la commande `\enquote{...}`, comme ceci : « Ceci est un exemple ».

## 1.1 Listes à puces et numérotées

Les listes sont stylisées pour une meilleure lisibilité.

### Exemple de liste à puces (itemize)

- Premier élément de la liste.
- Deuxième élément, qui peut s'étendre sur plusieurs lignes si nécessaire pour démontrer l'indentation.
- Troisième élément avec une sous-liste :
  - Sous-élément A.
  - Sous-élément B.

### Exemple de liste numérotée (enumerate)

1. Le premier pas est toujours le plus important.
2. Le deuxième suit logiquement.
3. Et ainsi de suite, avec une numérotation claire et en gras.

## 1.2 Structure des Titres

La classe définit un style particulier pour les sections, sous-sections et sous-sous-sections, comme vous pouvez le constater tout au long de ce document. Voici un exemple de la hiérarchie.

### 1.2.1 Ceci est un sous-sous-titre

Le style est plus discret pour indiquer un niveau de détail inférieur.

## 2 Mathématiques et Environnements Scientifiques

La classe intègre des outils puissants pour la rédaction de contenu scientifique, notamment les mathématiques et les théorèmes.

### 2.1 Équations et Formules

Les équations sont numérotées par section. Voici les équations de Maxwell, un exemple classique utilisant l'environnement `align`.

$$\nabla \cdot \mathbf{E} = \frac{\rho}{\varepsilon_0} \quad (2.1)$$

$$\nabla \cdot \mathbf{B} = 0 \quad (2.2)$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \quad (2.3)$$

$$\nabla \times \mathbf{B} = \mu_0 \left( \mathbf{J} + \varepsilon_0 \frac{\partial \mathbf{E}}{\partial t} \right) \quad (2.4)$$

#### Information

L'équation de Gauss (2.1) est fondamentale en électromagnétisme.

### 2.2 Théorèmes, Définitions et Remarques

Des environnements prédéfinis permettent de structurer le discours scientifique.

**Définition 2.1 (Groupe)** *Un groupe est un ensemble non vide  $G$  muni d'une loi de composition interne  $*$  qui est associative, admet un élément neutre et pour laquelle chaque élément admet un symétrique.*

**Théorème 2.1 (Théorème de Lagrange)** *Si  $H$  est un sous-groupe d'un groupe fini  $G$ , alors l'ordre de  $H$  divise l'ordre de  $G$ .*

**Exemple 1** *L'ensemble des entiers relatifs  $\mathbb{Z}$  muni de l'addition est un groupe.*

**Remarque 2.1** *Toutes ces boîtes partagent un style cohérent pour une lecture agréable.*

### 2.3 Unités Scientifiques

Le package `siunitx` est configuré pour le français.

### 1. Utilisation de `siunitx`

La constante de Planck est d'environ  $6,626 \times 10^{-34}$ . La vitesse de la lumière dans le vide est  $c = 299\,792\,458 \text{ m s}^{-1}$ .

## 3 Éléments Visuels : Figures et Tableaux

### 3.1 Insertion de Figures

La commande personnalisée `\insererfigure` simplifie l'ajout d'images encadrées.



**FIGURE 3.1** – Ceci est le logo principal, inséré avec notre commande personnalisée `\insererfigure`.

Pour des besoins plus complexes, comme les sous-figures, les environnements standards fonctionnent toujours.



**FIGURE 3.2** – Exemple de figure avec deux sous-figures utilisant le package `subcaption`.

### 3.2 Création de Tableaux

Les tableaux sont stylisés avec `booktabs` pour un rendu professionnel.

**TABLE 3.1** – Comparaison des caractéristiques de différents langages.

Langage	Caractéristique principale	Typage	Année
Python	Simplicité et lisibilité	Dynamique	1991
Java	« Write once, run anywhere »	Statique	1995
C++	Performance et contrôle système	Statique	1985
Populaires			



## 4 Boîtes d'Information et Listings de Code

### 4.1 Boîtes d'Information

Plusieurs types de boîtes colorées sont disponibles pour mettre en évidence certaines informations.

#### Résultat

C'est la fin de l'expérience. Le **résultat** est positif et confirme notre hypothèse de départ.

#### Comparaison

Par **comparaison**, l'approche A est 50% plus rapide que l'approche B, mais consomme plus de mémoire.

#### Observation

Une **observation** importante : le système devient instable lorsque la température dépasse 100 °C.

#### Attention

**Attention** : ne modifiez jamais les fichiers du noyau directement, au risque de corrompre le système.

### 4.2 Listings de Code

L'environnement `codeboxlang` permet d'afficher du code avec une coloration syntaxique adaptée au langage.

#### 1. Exemple de code Python

## PYTHON

```

1      # Simple script pour saluer le monde
2      def say_hello(name):
3          """
4          Cette fonction affiche un message de salutation.
5          """
6          print(f"Hello, {name}!")
7
8      if __name__ == "__main__":
9          say_hello("World")

```

## 2. Exemple de code Java

## JAVA

```

1      // Fichier: HelloWorld.java
2      public class HelloWorld {
3          /**
4           * Le point d'entrée du programme.
5           */
6          public static void main(String[] args) {
7              System.out.println("Hello, World from Java!");
8          }
9      }

```

## 3. Exemple de requête SQL

## SQL

```
1      -- Sélectionner les utilisateurs actifs
2      SELECT
3      user_id,
4      user_name,
5      registration_date
6      FROM
7      users
8      WHERE
9      is_active = 1
10     ORDER BY
11     registration_date DESC;
```

## 5 Nomenclature et Références

---

### 5.1 Nomenclature

Les termes définis dans le texte avec `\nomenclature` sont rassemblés ici.

### 5.2 Références et Hyperliens

Le package `hyperref` est configuré pour les liens internes et externes.

- Un lien interne vers la section sur les mathématiques : voir section ??.
- Un lien interne vers la figure des logos : voir figure ??.
- Un lien externe vers le site du projet LaTeX : <https://www.latex-project.org/>.

## 6 Conclusion

---

Ce document a exploré avec succès une grande partie des fonctionnalités de la classe `rapport.cls`. De la mise en page à la typographie mathématique, en passant par les éléments graphiques et les extraits de code, cette classe fournit un cadre robuste et esthétique pour la rédaction de rapports professionnels et académiques.

**Mise en perspective** Il est évident que la classe `rapport.cls` est un outil puissant pour la rédaction de rapports professionnels et académiques, offrant une gamme complète de fonctionnalités pour assurer une mise en page esthétique et structurée. De plus, son intégration des éléments mathématiques permet d'obtenir des documents bien formatés dans ce domaine spécifique. Enfin, la présence d'extraits de code, tels que les listings LaTeX ou le Python, offre une avance supplémentaire en termes de clarté et de facilité à la compréhension pour un large éventail de lecteurs.

# Merci d'avoir consulté cette présentation